Quick answers to common problems

WordPress 3 Cookbook

Over 100 recipes to help you enhance your WordPress site!

Ric Shreves
Jean-Baptiste Jung

[PACKT] PUBLISHING open source*
community experience distilled

WordPress 3 Cookbook

Over 100 recipes to help you enhance your WordPress site!

Ric Shreves

Jean-Baptiste Jung

[PACKT] open source
PUBLISHING
community experience distilled

BIRMINGHAM - MUMBAI

WordPress 3 Cookbook

Copyright © 2011 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: July 2009

Second published: December 2011

Production Reference: 1071211

Livery Place
35 Livery Street
Birmingham B3 2PB, UK.

ISBN 978-1-84951-460-6

`www.packtpub.com`

Cover Image by J.Blaminsky (`jarek@jblaminsky.com`)

Credits

Author
Ric Shreves
Jean-Baptiste Jung

Reviewers
Shameer C
John Eckman

Acquisition Editor
Usha Iyer

Development Editor
Susmita Panda

Technical Editor
Merwine Machado

Project Coordinator
Kushal Bhardwaj

Proofreader
Aaron Nash

Indexer
Rekha Nair

Production Coordinator
ArvindKumar Gupta

Cover Work
ArvindKumar Gupta

About the Author

Ric Shreves is one of the founding partners of water&stone, an interactive agency that specialises in open source web content management systems. He has been building CMS websites for over 10 years, and during that time, he has been involved in projects for a number of global brands, including BASF, BearingPoint, Colgate-Palmolive, Tesco Lotus, CBRichard Ellis, Mercy Corps, and many others. He has published a number of books on open source in general and on open source content management systems in particular. Past work includes books on Mambo, Drupal, Joomla!, and Ubuntu. This is his first book on the WordPress CMS.

Ric lives in Bali with his wife and business partner, Nalisa.

I would like to thank Packt Publishing for giving me the opportunity to work on this WordPress title. With the advent of WordPress 3, the system has come into its own as a CMS and I am proud to be able to do something to help users get the most out of the system.

I would also like to thank my partner, Nalisa, for her patience and support during the writing of this text.

About the Reviewer

Shameer C is a software engineer and database administrator from Kerala, India. He started his career as a web developer after completing B-tech in computer science. He has experience in various programming languages such as PHP, Javascript, HTML5, Ruby, and so on. On the database side he mainly uses MySQL. He has contributed to Aura PHP, which is an enterprise-level php 5.3 framework and customized Toto, a flat file blogging engine in ruby. Occasionally he writes articles on his blog at `http://shameerc.com`. You can reach him at `twitter @shameerc.`

Shameer is currently working for Qburst Technologies, a company with expertise in both mobile and web application development on various platforms.

I would like to thank Packt Publishing for giving me an opportunity to work on their project, and all my family members, friends, especially Hari K T, for inspiring and encouraging me throughout my career.

John Eckman has more than a decade of experience in designing and building web applications for organizations ranging from small non-profit organizations to Fortune 500 enterprises. Currently a Digital Strategist with ISITE Design, he often works with clients to develop and execute complex web applications, but not exclusively with open source platforms.

John received a Bachelor of Arts from Boston University, a Masters in Information Systems from Northeastern University, and a Ph.D. from the University of Washington, Seattle. He is an active contributor to a number of open source communities, a founding organizer of WordCamp Boston 2010 and 2011, and the lead developer of the WPBook plugin for WordPress. He blogs at `www.openparenthesis.org` and tweets as `@jeckman`.

I'd like to thank the broader WordPress community — users and developers — without whom none of this would be possible.

---

www.PacktPub.com

Support files, eBooks, discount offers and more

You might want to visit `www.PacktPub.com` for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at `www.PacktPub.com` and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at `service@packtpub.com` for more details.

At `www.PacktPub.com`, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.

PACKTLiB™

`http://PacktLib.PacktPub.com`

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

Why Subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print and bookmark content
- On demand and accessible via web browser

Free Access for Packt account holders

If you have an account with Packt at `www.PacktPub.com`, you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

Table of Contents

Preface 1

Chapter 1: The WordPress Cook's Tools 5

- Introduction 5
- Managing media files with the Media Library 6
- Modifying theme files with the built-in Theme Editor 10
- Modifying plugin files with the built-in Plugin Editor 13
- Managing users 14
- Gaining control over user roles and permissions 18
- Setting up editorial workflow 21
- Importing and exporting content 24
- Installing and using Jetpack 26
- Enabling the toolbar for users and administrators 29

Chapter 2: Installing and Customizing Themes 31

- Introduction 32
- Installing a theme 32
- Creating a child theme 36
- Modifying your theme colors 38
- Modifying your theme fonts 40
- Creating and integrating a favicon 43
- Adding a custom logo 45
- Customizing the login page 48
- Using conditional tags to control content display 52
- Using multiple page templates 56
- Using post formats 59
- Creating a custom 404 error page 63
- Using a static page as a homepage 64
- Adding custom styles to your theme 66
- Making your site mobile device friendly 67

Chapter 3: Working with Plugins and Widgets 71
Introduction 71
Installing plugins 72
Installing widgets 77
Adding widget areas to your themes 79
Creating your own widget 82
Modifying core widgets 85
Displaying tabs on your sidebar 92
Using conditional tags to control widget display 94
Displaying widgets inside of posts and pages 97
Chapter 4: Customizing Content Display 101
Introduction 102
Accessing posts within the WordPress loop 103
Retrieving posts from a specific category 105
Getting a specific number of posts 107
Retrieving posts by date 108
Displaying posts published today 110
Displaying posts published exactly one year ago 111
Using multiple loops 112
Accessing post data outside of the WordPress loop 114
Accessing permalinks outside the loop 118
Displaying thumbnails on your homepage 119
Alternating background colors on post lists 121
Displaying posts in two columns 123
Save time by using WordPress shortcodes 125
Enabling the use of shortcodes in widgets 126
Adding notes to your posts 128
Adding tags to your pages 130
Chapter 5: Building Interactivity and Community 133
Introduction 134
Improving navigation with a paginator 134
Highlighting searched text in search results 137
Integrating a forum into your site 139
Adding social bookmarking buttons to your theme 144
Aggregating RSS content 146
Integrating Feedburner into your site 150
Displaying a retweet button on your posts 152
Getting more comments with the Subscribe to Comments Reloaded plugin 155
Remove the nofollow attribute to motivate users to leave comments 156
Provide recognition to your top contributors 158

Displaying author-related information on posts 160
Displaying the author's avatar on posts 163
Allowing multiple authors on posts 164
Displaying a list of all of the authors 166
Creating community with BuddyPress 169
Adding a simple gallery to your site 174
Bringing Facebook functionality into your site 178
Integrating a Twitter stream into your site 180
Chapter 6: Implementing Online Sales and Advertising 187
Introduction 187
Integrating Adsense 188
Displaying ads anywhere in your posts by using WordPress shortcodes 192
Managing ad visibility 194
Inserting ads into your RSS feeds 198
Showing your site stats to find advertisers 200
Enhancing your Advertise page by adding Paypal subscriptions 206
Managing your advertising space with an ad manager 207
Adding a shopping cart to your site 213
Chapter 7: Making an SEO Friendly Site 217
Introduction 217
Making your site visible to search engines 218
Optimizing your permalinks for SEO 220
Migrating your permalinks safely 224
Adding redirects for changed URLs 225
Creating meta descriptions for your posts and pages 226
Avoiding duplicate content with a robots.txt file 229
Pinging third-party services 231
Enhancing site indexing with XML sitemaps 233
Using Google's and Bing's Webmaster Tools 237
Improving SEO with the SEO Ultimate plugin 241
Chapter 8: Enhancing Usability and Accessibility 245
Introduction 245
Creating print-friendly pages 246
Extending WordPress search 249
Enhancing navigation with breadcrumbs 252
Stopping SPAM 256
Optimizing performance with cache management 258
Displaying a login form 262
Displaying related posts 265
Creating a Featured Posts block 268

Adding a sitemap for your site visitors 271
Creating a better tag cloud 274
Adding lightboxes for your photos 276
Chapter 9: Managing Maintenance and Improving Security 279
Introduction 280
Creating a manual backup of your database 280
Creating an automatic backup with WP DB Backup 282
Restoring a MySQL backup 284
Creating backups of your WordPress files 286
Removing the WordPress version information from your theme files 287
Getting rid of the Administrator account 288
Protecting against brute force log in attempts 289
Denying access to unneeded hints 290
Adding another layer of protection with HTTP authentication 292
Restricting access to the wp-admin directory by using the IP address 294
Testing your site security 296
Reducing SPAM by selectively blocking comment posting 301
Index 303

Preface

WordPress 3 Cookbook will help you get the most from version 3 of the popular WordPress CMS. The book is focused on showing how to achieve the most commonly desired system modifications and customizations, with an emphasis on enhancing themes and content presentation. Other chapters look at the practicalities of owning a WordPress site, including SEO, advertising, online sales, and site maintenance.

What this book covers

Chapter 1, *The WordPress Cook's Tools*, explains the basic tools and options that are built into the default WordPress CMS and discusses simple enhancements that can extend the base functionality.

Chapter 2, *Installing and Customizing Themes*, covers how to install and customize themes for your WordPress site. The chapter covers important concepts such as creating child themes and how to use conditional tags to control theme output.

Chapter 3, *Working with Plugins and Widgets*, dives into the WordPress CMS plugin system. The contents cover how to install new plugins, as well as how to customize plugins and widgets.

Chapter 4, *Customizing Content Display*, deals with issues related to WordPress themes. The chapter includes both the basics of how to modify your existing theme, and more advanced theming topics that give you the ability to customize any WordPress theme.

Chapter 5, *Building Interactivity and Community*, examines how you can use social media and social sharing tools to increase participation on your site and help bring your content to the attention of potential new visitors.

Chapter 6, *Implementing Online Sales and Advertising*, we look at how to implement advertising and online sales inside the WordPress CMS. Topics range from basics like adding Google AdSense, to more advanced topics like adding a shopping cart and PayPal payment.

Chapter 7, *Making an SEO Friendly Site*, takes on the topic of search engine optimization. The recipes show how to enhance your site's SEO and how to improve your site's chances of ranking well on the search engines.

Chapter 8, *Enhancing Usability and Accessibility*, covers accessibility and usability issues, with recipes that show how to make your site more accessible and usable by a wider range of visitors.

Chapter 9, *Managing Maintenance and Improving Security*, discusses the issues related to owning a WordPress site, including how to manage updates and upgrades and improve WordPress CMS security.

What you need for this book

Technically, all you need to have in order to benefit from this book is access to an installation of Version 3 of the WordPress open source CMS. That said, you will also find it useful to have the following:

- Your favorite code editor, whether it's a basic text editor or something such as Dreamweaver
- An FTP program, or other means of moving files to and from the server where you WordPress installation is located

Who this book is for

The WordPress 3 Cookbook was intended for a wide audience of potential WordPress users, from casual website owners who simply want to know how to do a bit more with their site to developers who are looking for tried and true solutions to common problems.

Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text are shown as follows: " The Breadcrumbs Plus plugin enables the `breadcrumbs_plus ()` function."

A block of code is set as follows:

```
/**Time to register the widget*/

add_action( 'widgets_init', create_function('', 'return
   register_widget("Meta_Mod");') );

?>
```

New terms and important words are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this:" Click on the **Save** button ".

Warnings or important notes appear in a box like this.

Tips and tricks appear like this.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book—what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to `feedback@packtpub.com`, and mention the book title via the subject of your message.

If there is a book that you need and would like to see us publish, please send us a note in the **SUGGEST A TITLE** form on `www.packtpub.com` or e-mail `suggest@packtpub.com`.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on `www.packtpub.com/authors`.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

Downloading the example code

You can download the example code files for all Packt books you have purchased from your account at `http://www.PacktPub.com`. If you purchased this book elsewhere, you can visit `http://www.PacktPub.com/support` and register to have the files e-mailed directly to you.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting `http://www.packtpub.com/support`, selecting your book, clicking on the **errata submission form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from `http://www.packtpub.com/support`.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at `copyright@packtpub.com` with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at `questions@packtpub.com` if you are having a problem with any aspect of the book, and we will do our best to address it.

1

The WordPress Cook's Tools

In this chapter you will learn about:

- Managing media files with the Media Library
- Modifying theme files with the built-in Theme Editor
- Modifying plugin files with the built-in Plugin Editor
- Managing users
- Gaining control over user roles and permissions
- Setting up editorial workflow
- Importing and exporting content
- Installing and using Jetpack
- Enabling the toolbar for users and administrators

Introduction

This chapter serves as an introduction to the basic tools and features that are part of your WordPress administration system. All of the topics discussed in this chapter relate to fundamental functionality needed to use the WordPress CMS. Most of the chapter is focused on tools that are included in the default WordPress system.

Understanding the tools discussed in this chapter is essential to understand how to get the most out of your WordPress site. There's a lot you can do with WordPress, even without installing additional plugins and custom themes. In this chapter, we look at how the basic tools enable you to work with users, set permissions, and workflow, and how you can even modify plugins and themes – all from within the WordPress administration interface and without the need for specialized or external tools.

Managing media files with the Media Library

WordPress includes a tool designed to help you deal with the media files for your website. The tool, called appropriately the Media Library, allows you to view and manage all your media files (images, videos, and so on) in one place.

Once you've added files to your Media Library, they are available to you as you work with the posts and pages of your site. The key advantage of using the Library is that you can work with images in bulk, uploading or deleting multiple images at one time.

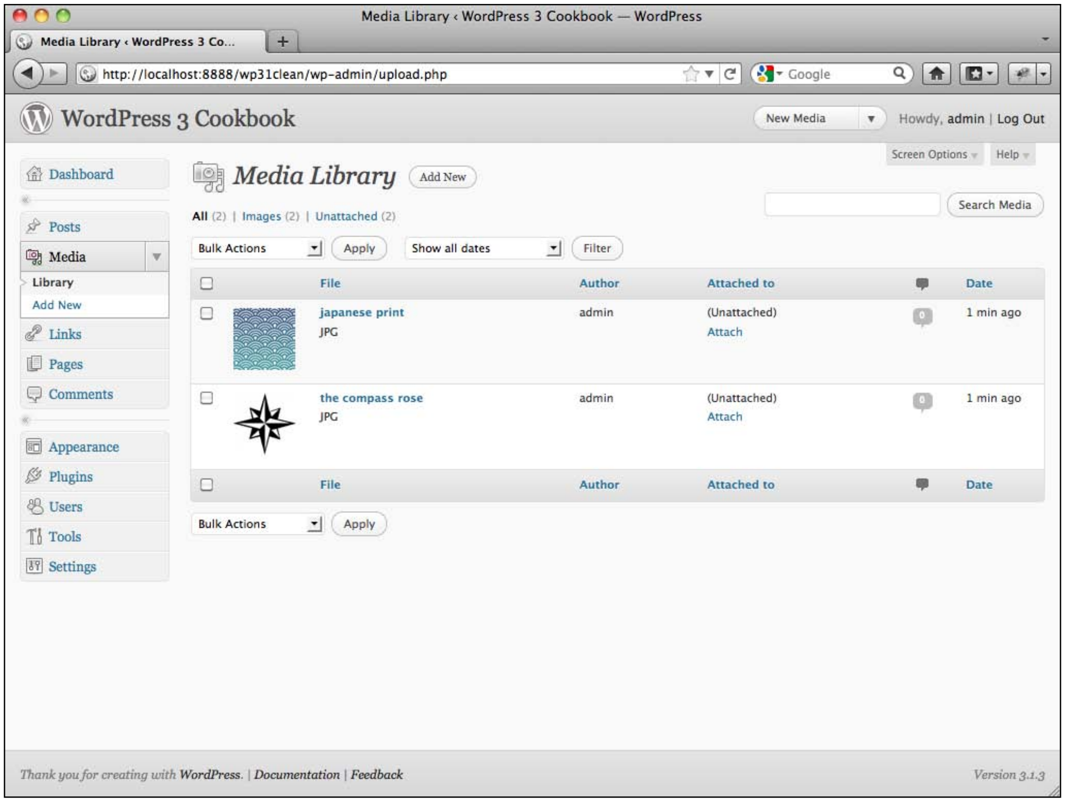
In this recipe, we cover the basics of working with this useful tool, including adding, editing, and deleting files.

Getting ready

Everything you need for this recipe is located inside the dashboard of your WordPress site.

How to do it...

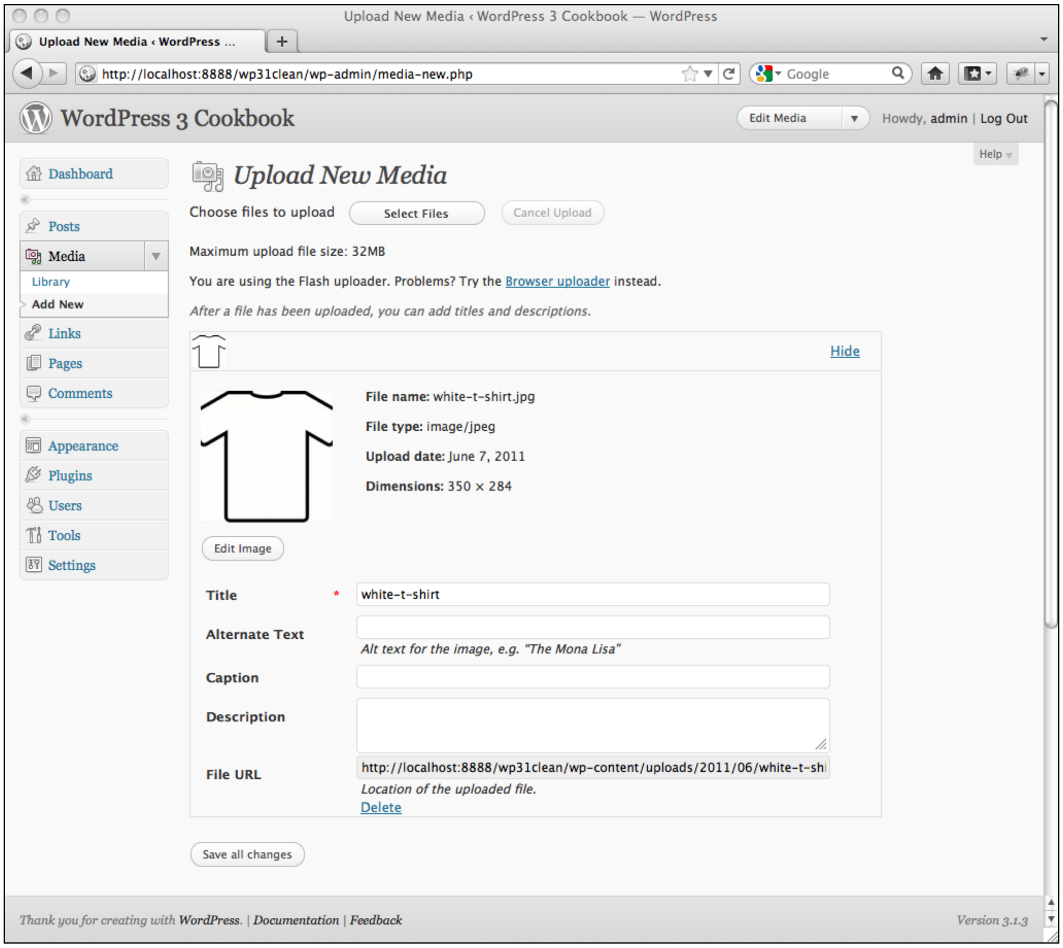
1. Log in to your WordPress **Dashboard**.
2. Click on the **Media** menu.
3. To view the files in the Library, click on the **Library** option and you'll see the existing files, if any. The following screenshot shows you a typical view:


Media Library ‹ WordPress 3 Cookbook — WordPress
Media Library ‹ WordPress 3 Co...
http://localhost:8888/wp31clean/wp-admin/upload.php
Google
WordPress 3 Cookbook
New Media
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Library
Add New
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Media Library
Add New
Search Media
All (2) | Images (2) | Unattached (2)
Bulk Actions
Apply
Show all dates
Filter
Bulk Actions
Apply
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

☐	File	Author	Attached to	💬	Date
☐	japanese print JPG	admin	(Unattached) Attach	0	1 min ago
☐	the compass rose JPG	admin	(Unattached) Attach	0	1 min ago

4. To add a new media file to your **Media Library**, click on **Add New**.
5. On the screen that loads in your browser, click on the **Select Files** button and the system will show you a pop-up that lets you to select the media files from your hard drive.
6. Once you locate the file you want, select it, then click the **Upload** button and the system will add the file to the **Media Library**. As it uploads, you will see a status bar showing you the progress.

7. Once the upload is complete you can view the file in the library and edit the details, if you so desire. The following screenshot shows you the **Upload New Media** screen, where you can edit the image info. Once you finish with your edits, click on **Save all changes**.


Upload New Media ‹ WordPress 3 Cookbook — WordPress
Upload New Media ‹ WordPress ...
http://localhost:8888/wp31clean/wp-admin/media-new.php
Google
WordPress 3 Cookbook
Edit Media
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Library
Add New
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Upload New Media
Choose files to upload
Select Files
Cancel Upload
Hide
Edit Image
Delete
Save all changes
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

Maximum upload file size: 32MB

You are using the Flash uploader. Problems? Try the Browser uploader instead.

After a file has been uploaded, you can add titles and descriptions.

File name: white-t-shirt.jpg

File type: image/jpeg

Upload date: June 7, 2011

Dimensions: 350 × 284

Title * white-t-shirt

Alternate Text

Alt text for the image, e.g. "The Mona Lisa"

Caption

Description

File URL http://localhost:8888/wp31clean/wp-content/uploads/2011/06/white-t-shi

Location of the uploaded file.

> There are two uploaders available: the Flash uploader and the Browser uploader. The Flash uploader allows you to select multiple files at once, while the Browser uploader allows you to upload only one file at a time. While the Flash uploader can be faster and more convenient, on some systems you may experience some difficulties using it. If you have any issues with the Flash uploader, simply choose the Browser uploader; it's slower, but it's very reliable.

To delete files from the **Media Library**, carry out the following steps:

1. When inside the **Media Library**, simply hover the mouse over an item and the **Edit**, **Delete Permanently**, and **View** buttons will be appear.
2. Click on **Delete Permanently** and the system will prompt you for confirmation.
3. If you wish to remove the file, click on **OK** in the pop-up and the system will delete the file.

For bulk media deletion, carry out the following steps:

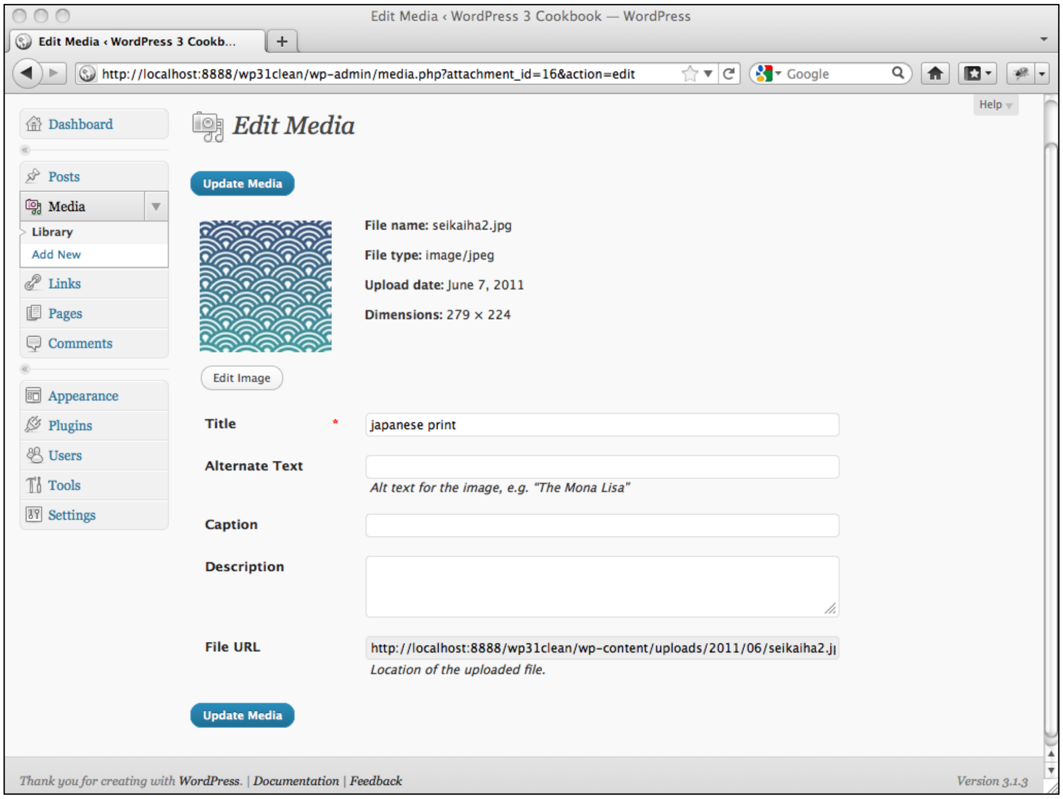
1. Go to the **Media Library.**
2. Select the checkboxes immediately to the left of the files you wish to delete.
3. Select the option **Delete Permanently** from the **Bulk Actions** drop-down list (located above the list of files)
4. Click on the **Apply** button.

> Be careful – when using bulk deletion, there is no confirmation dialogue! Once you click the **Apply** button the system will immediately delete all the files you have selected.

Editing files in the **Media Library** is limited to modifying the meta information associated with the file; you cannot actually edit the media file itself. To edit existing file's information, carry out the following steps:

1. Access the **Media Library**.
2. Hover the mouse over the item you'd like to edit and click on the **Edit** link that appears.

3. In the page that opens, you can define the file settings. The following screenshot shows the **Edit Media** page:


Edit Media ‹ WordPress 3 Cookbook — WordPress
Edit Media ‹ WordPress 3 Cookb...
http://localhost:8888/wp31clean/wp-admin/media.php?attachment_id=16&action=edit
Google
Help
Dashboard
Posts
Media
Library
Add New
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Edit Media
Update Media
Edit Image
Update Media

File name: seikaiha2.jpg
File type: image/jpeg
Upload date: June 7, 2011
Dimensions: 279 × 224

Title * japanese print

Alternate Text
Alt text for the image, e.g. "The Mona Lisa"

Caption

Description

File URL http://localhost:8888/wp31clean/wp-content/uploads/2011/06/seikaiha2.j
Location of the uploaded file.

Thank you for creating with WordPress. | *Documentation* | *Feedback* *Version 3.1.3*

4. Click on the **Update Media** button when you're done and the system will save your changes.

Modifying theme files with the built-in Theme Editor

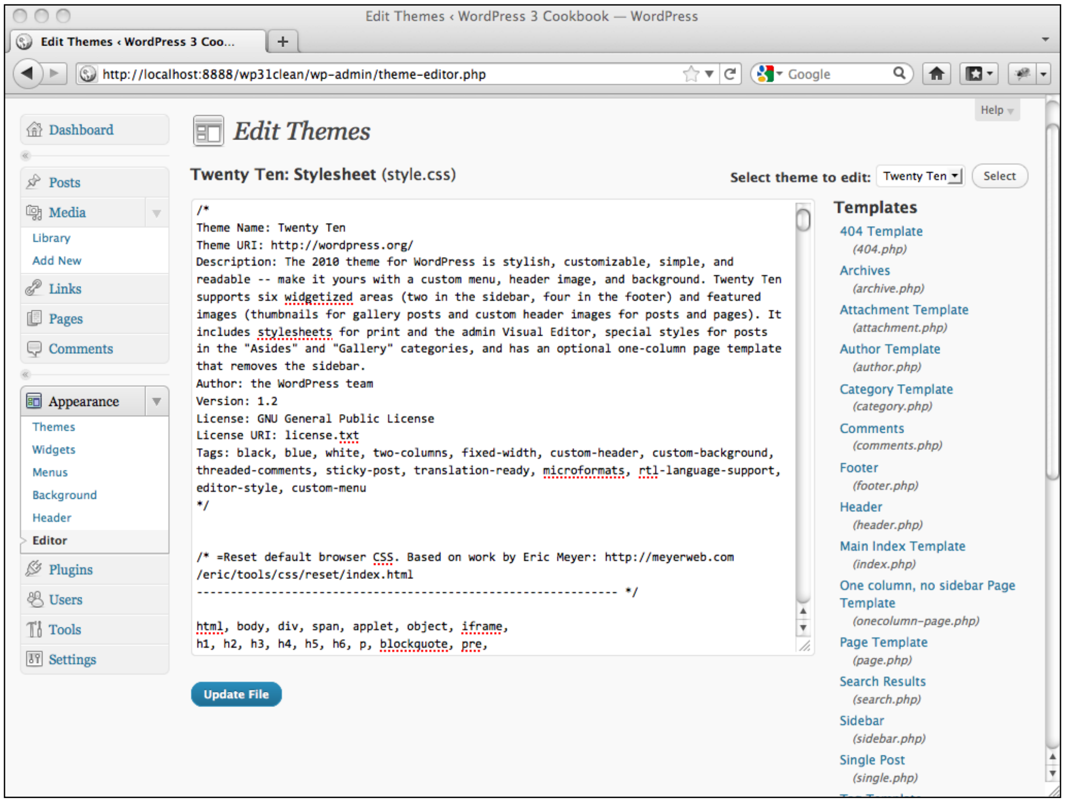
As you are probably aware, the appearance of your WordPress site is dictated by the theme you use. Themes themselves are comprised of a number of files, typically a mix of PHP and CSS files. Editing the files in your theme can be handled in one of two ways: either with a third party editor, or with WordPress' built-in Theme Editor. In this recipe, we introduce the basics of working with the Theme Editor.

Getting ready

Everything you need to complete this recipe is located inside your WordPress dashboard.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.
3. Click on the option **Editor**.
4. By default, the system will load one of the files from the active theme, as seen in the following screenshot. If you wish to change the view to edit a different file, simply click on the name of the file in the right hand column.
5. Make your changes.
6. When you're done, click on the **Update File** button to save your modifications.


Edit Themes ‹ WordPress 3 Cookbook — WordPress
Edit Themes ‹ WordPress 3 Coo...
http://localhost:8888/wp31clean/wp-admin/theme-editor.php
Google
Help
Dashboard
Posts
Media
Library
Add New
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Background
Header
Editor
Plugins
Users
Tools
Settings
Edit Themes
Select theme to edit:
Twenty Ten
Select
Templates
404 Template
(404.php)
Archives
(archive.php)
Attachment Template
(attachment.php)
Author Template
(author.php)
Category Template
(category.php)
Comments
(comments.php)
Footer
(footer.php)
Header
(header.php)
Main Index Template
(index.php)
One column, no sidebar Page Template
(onecolumn-page.php)
Page Template
(page.php)
Search Results
(search.php)
Sidebar
(sidebar.php)
Single Post
(single.php)
Update File

Twenty Ten: Stylesheet (style.css)

```
/*
Theme Name: Twenty Ten
Theme URI: http://wordpress.org/
Description: The 2010 theme for WordPress is stylish, customizable, simple, and
readable -- make it yours with a custom menu, header image, and background. Twenty Ten
supports six widgetized areas (two in the sidebar, four in the footer) and featured
images (thumbnails for gallery posts and custom header images for posts and pages). It
includes stylesheets for print and the admin Visual Editor, special styles for posts
in the "Asides" and "Gallery" categories, and has an optional one-column page template
that removes the sidebar.
Author: the WordPress team
Version: 1.2
License: GNU General Public License
License URI: license.txt
Tags: black, blue, white, two-columns, fixed-width, custom-header, custom-background,
threaded-comments, sticky-post, translation-ready, microformats, rtl-language-support,
editor-style, custom-menu
*/


/* =Reset default browser CSS. Based on work by Eric Meyer: http://meyerweb.com
/eric/tools/css/reset/index.html
-------------------------------------------------------------- */

html, body, div, span, applet, object, iframe,
h1, h2, h3, h4, h5, h6, p, blockquote, pre,
```

How it works...

The Theme Editor simply provides an editing interface for the files in the active theme. You can change to edit the files of a different theme by selecting the theme name from the combo box labeled **Select theme to edit**.

> Be cautious! Remember that when you edit a theme file in WordPress Theme Editor, you are editing the real file on the server. Once you press the **Update File** button, the file is saved and the previous version is erased. This is an issue of particular importance where the theme you are editing is the active theme on a live site.

There's more...

Although the Theme Editor is very convenient, you have to be careful with it.

> Depending on your web hosting environment, you may experience problems using the Theme Editor. Accordingly, it is essential that you have a back up of your WordPress files before you begin working.

- Best practice is to create a backup of your theme before editing. If you have made a modification and later would like to undo the modification, you need to have a backup of the previous version of the file.
- Use the Theme Editor only if you're sure about what you're doing. If you're editing your current theme and make a programming error (for example, a PHP syntax error), it is possible that your site will stop functioning until you correct the error.
- Sometimes, a programming mistake can even result in you losing access to the Theme Editor. While this is quite a rare case (it mostly happens when you make a code mistake in the `functions.php` file), the problem is serious. You will most likely need to have a backup of your theme (as well as an FTP connection to your server) to sort out this problem.

See also

- *Chapter 2*, *Installing and Customizing Themes*, covers WordPress themes in more detail

Modifying plugin files with the built-in Plugin Editor

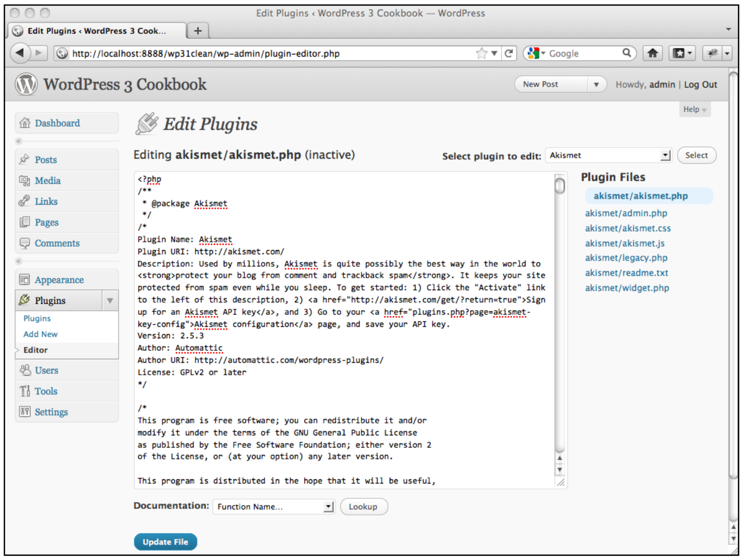
Plugins are a vital part of your WordPress site, adding much of the key functionality to your site. Just as we saw with the Theme Editor in the previous recipe, you can edit plugin files directly from within the WordPress dashboard, without the need of a third party editor. In this recipe, we introduce the basics of working with the Plugin Editor.

Getting ready

Everything you need to complete this recipe is located inside your WordPress dashboard.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Plugins** menu.
3. Click on the option **Editor**.
4. By default, the system will display one of the files from the first available plugin, as seen in the next screenshot. If you wish to change the view to edit a different file for the plugin, simply click the name of the file in the right hand column. To change to a different plugin, select one from the combo box labeled **Select plugin to edit**.
5. Make your changes.
6. When you're done, click on the **Update File** button to save your modifications.


Edit Plugins ‹ WordPress 3 Cookbook — WordPress
http://localhost:8888/wp31clean/wp-admin/plugin-editor.php
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Settings
Edit Plugins
Editing akismet/akismet.php (inactive)
Select plugin to edit: Akismet
Select
Plugin Files
akismet/akismet.php
akismet/admin.php
akismet/akismet.css
akismet/akismet.js
akismet/legacy.php
akismet/readme.txt
akismet/widget.php
Documentation: Function Name...
Lookup
Update File

```
<?php
/**
 * @package Akismet
 */
/*
Plugin Name: Akismet
Plugin URI: http://akismet.com/
Description: Used by millions, Akismet is quite possibly the best way in the world to
<strong>protect your blog from comment and trackback spam</strong>. It keeps your site
protected from spam even while you sleep. To get started: 1) Click the "Activate" link
to the left of this description, 2) <a href="http://akismet.com/get/?return=true">Sign
up for an Akismet API key</a>, and 3) Go to your <a href="plugins.php?page=akismet-
key-config">Akismet configuration</a> page, and save your API key.
Version: 2.5.3
Author: Automattic
Author URI: http://automattic.com/wordpress-plugins/
License: GPLv2 or later
*/

/*
This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
```

How it works...

The built-in Plugin Editor works in exactly the same way as the Theme Editor. When a file is modified and saved, the modifications are written directly in the source file—there's no copy or backup.

> The system provides a link to documentation for the plugin. If you look below the editing window in the previous screenshot, you can see a combo box labeled **Documentation**. Select the appropriate file from the list then click **Lookup** to view the documentation.

There's more...

The Plugin Editor is a very useful tool; however, it can also create problems if used improperly.

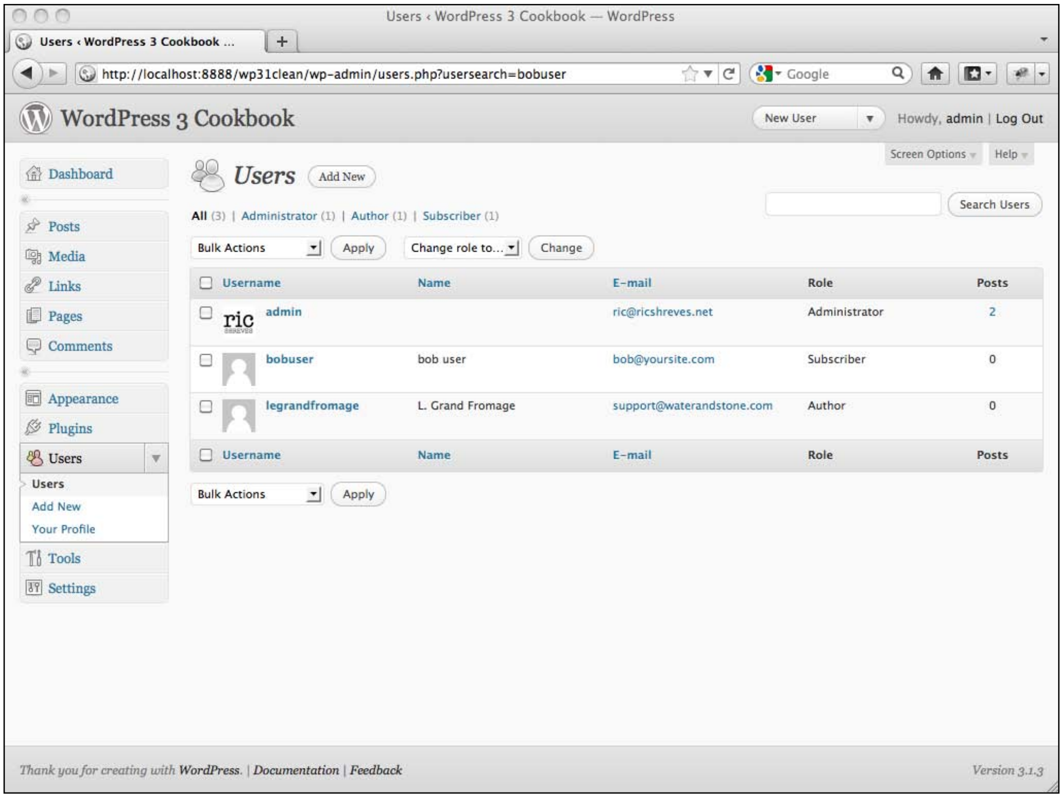
- Unless you're very sure about what you're doing, always deactivate the plugin before editing
- Always have a backup of the plugin you're editing, as the Plugin Editor does not save any revisions
- If—after editing a plugin—your site does not function correctly, deactivate the plugin, and upload your plugin files backup to your `wp-content/plugins/yourplugin` directory

See also

- *Chapter 3*, *Working with Plugins and Widgets*, covers WordPress plugins in more detail

Managing users

All WordPress sites include a combination of public and registered users. Registered users can be assigned to various roles that give access to different features of the site. WordPress includes a feature that enables you to manage the registered users of your website the Users Manager, as shown in the following screenshot:


Users ‹ WordPress 3 Cookbook — WordPress
Users ‹ WordPress 3 Cookbook ...
http://localhost:8888/wp31clean/wp-admin/users.php?usersearch=bobuser
Google
WordPress 3 Cookbook
New User
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Users
Add New
Your Profile
Tools
Settings
Users
Add New
Search Users
All (3) | Administrator (1) | Author (1) | Subscriber (1)
Bulk Actions
Apply
Change role to...
Change
Bulk Actions
Apply
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

Username	Name	E-mail	Role	Posts
admin		ric@ricshreves.net	Administrator	2
bobuser	bob user	bob@yoursite.com	Subscriber	0
legrandfromage	L. Grand Fromage	support@waterandstone.com	Author	0

In this recipe we introduce the Users Manager and the basics of creating, editing, and deleting registered users in WordPress.

Getting ready

Everything you need for this recipe can be found in the WordPress dashboard.

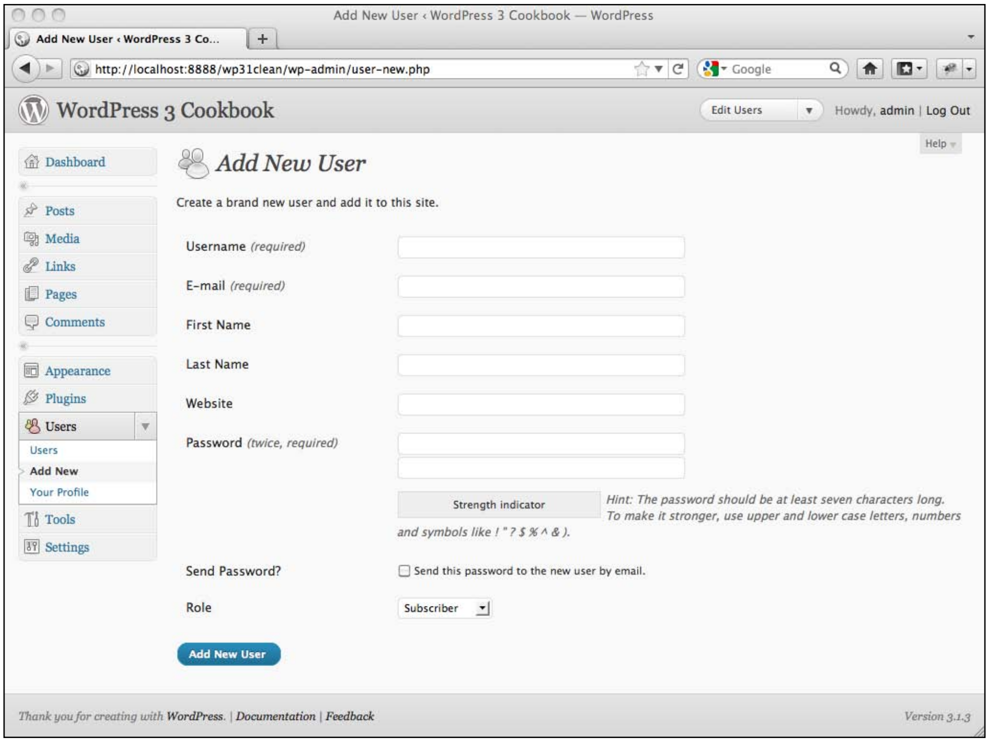
How to do it...

WordPress users manager allows you to add, edit, or delete user accounts. Let's learn how to do this in detail.

In order to add a new user, carry out the following steps:

1. Log in to the WordPress **Dashboard**.
2. Click on the menu labeled **Users**. The Users Manager, as shown in the previous screenshot, will load.
3. Click on **Add New**.

4. The **Add New User** screen in shown in the following screenshot. The only required fields are **Username**, **E-mail**, and **Password** (which must be entered twice). However, you should also check the **Role** control and make sure you are setting the right access privileges for the user.
5. Once done, click on the **Add New User** button.


Add New User ‹ WordPress 3 Cookbook — WordPress
Add New User ‹ WordPress 3 Co...
http://localhost:8888/wp31clean/wp-admin/user-new.php
Google
WordPress 3 Cookbook
Edit Users
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Users
Add New
Your Profile
Tools
Settings
Strength indicator
Add New User

Add New User

Create a brand new user and add it to this site.

Username *(required)*

E-mail *(required)*

First Name

Last Name

Website

Password *(twice, required)*

Hint: The password should be at least seven characters long. To make it stronger, use upper and lower case letters, numbers and symbols like ! " ? $ % ^ &).

Send Password? ☐ Send this password to the new user by email.

Role: Subscriber

Thank you for creating with WordPress. | Documentation | Feedback

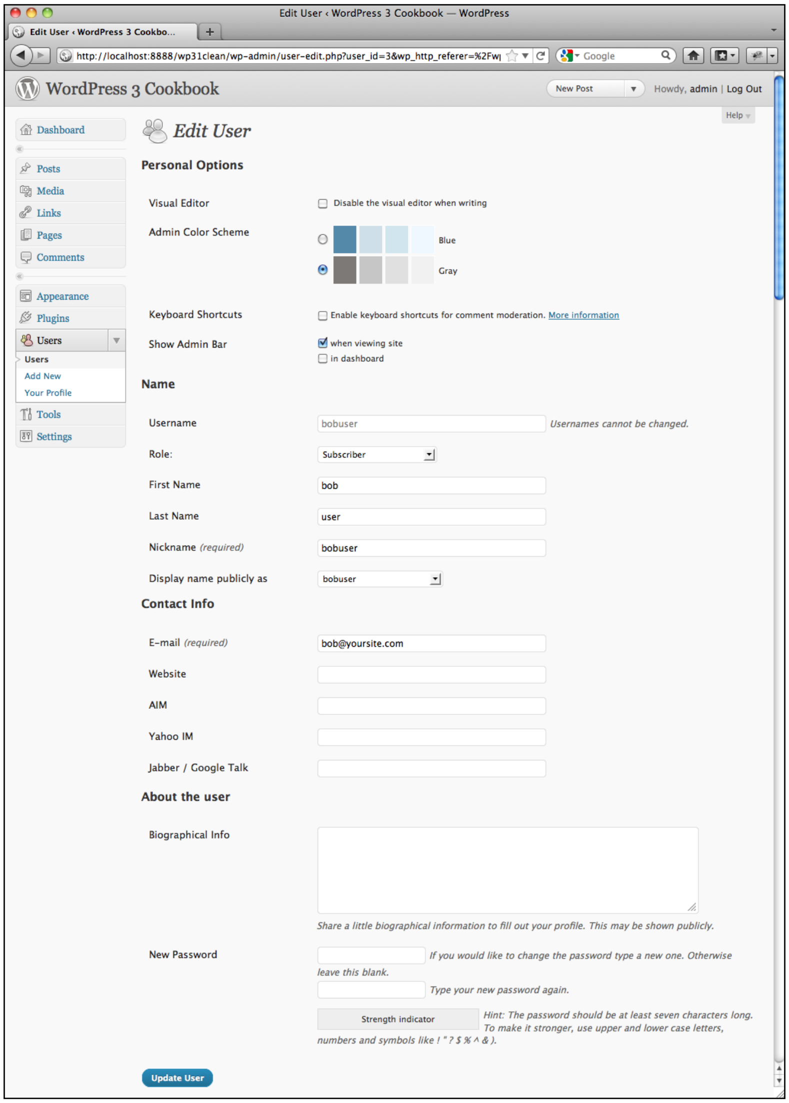
Version 3.1.3

In order to edit an existing user account, carry out the following steps:

1. Access the **Users** menu, as explained in the previous steps.
2. Find the name of the user you wish to edit and hover the mouse over the name; the **Edit** and **Delete** buttons will be displayed.
3. Click on the **Edit** button.
4. On the next page, as shown in the next screenshot, you can edit the following information about the user:
    - Enable/Disable **Visual Editor**
    - **Admin color scheme**
    - Enable/Disable **Keyboard Shortcuts**
    - User **Role**

- **First Name**, **Last Name**, and **Nickname**
- How the user name should be publicly displayed
- **Contact info**
- User bio
- Password

5. Make the changes you desire.
6. Click on the **Update User** button to save your modifications.


Edit User ‹ WordPress 3 Cookbook — WordPress
Edit User ‹ WordPress 3 Cookbo...
http://localhost:8888/wp31clean/wp-admin/user-edit.php?user_id=3&wp_http_referer=%2Fw
Google
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Users
Add New
Your Profile
Tools
Settings

Edit User

Personal Options

Visual Editor ☐ Disable the visual editor when writing

Admin Color Scheme ○ Blue ● Gray

Keyboard Shortcuts ☐ Enable keyboard shortcuts for comment moderation. More information

Show Admin Bar ☑ when viewing site ☐ in dashboard

Name

Username: bobuser — *Usernames cannot be changed.*

Role: Subscriber

First Name: bob

Last Name: user

Nickname *(required)*: bobuser

Display name publicly as: bobuser

Contact Info

E-mail *(required)*: bob@yoursite.com

Website:

AIM:

Yahoo IM:

Jabber / Google Talk:

About the user

Biographical Info:

Share a little biographical information to fill out your profile. This may be shown publicly.

New Password: *If you would like to change the password type a new one. Otherwise leave this blank.*

Type your new password again.

Strength indicator *Hint: The password should be at least seven characters long. To make it stronger, use upper and lower case letters, numbers and symbols like ! " ? $ % ^ &).*

Update User

In order to delete a user's account, carry out the following steps:

1. Access the Users Manager, as discussed previously.
2. Find the user you'd like to delete (a mini search engine is included on the top right of the page) and place the mouse cursor over his or her name. The **Edit** and **Delete** button will appear.
3. Click on the **Delete** button.
4. The system will prompt you to choose between deleting the user and all the content which he has provided (posts, comments, and so on) or deleting the user, but transferring the content to another author. Make your selection.
5. Click on **Confirm Deletion**.

> You can also delete multiple users simultaneously by selecting them on the main users manager page, then choosing **Delete** from the **Bulk Actions** combo box.

See also

- *Gaining control over user roles and permissions* section in this chapter
- *Setting up editorial workflow* section in this chapter

Gaining control over user roles and permissions

User permissions in WordPress are dictated by the role the user is assigned to. By default, the WordPress system includes five roles:

- Admin
- Editor
- Author
- Contributor
- Subscriber

The permissions associated with each role are fixed and cannot be edited without the use of a plugin. While there are several plugins that provide this functionality, in this recipe we take you through using the User Role Editor and show you how to both modify existing roles and how to create new ones.

To learn more about the default user roles and their capabilities, visit the WordPress Codex page on the subject at `http://codex.wordpress.org/Roles_and_Capabilities`

Getting ready

To execute this recipe, you will need to install the User Role Editor plugin. You will need to install this plugin before you can get started. Search for **User Role Editor** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

You can learn more about the plugin by visiting the developer's website at `http://www.shinephp.com/user-role-editor-wordpress-plugin/`

How to do it...

To edit an existing role, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Users** menu.
3. Click on the options **User Role Editor**.
4. On the page that loads, first select the role you wish to edit from the **Select Role** combo box.
5. Make the changes you desire.
6. Click on the **Update** button to save your changes.

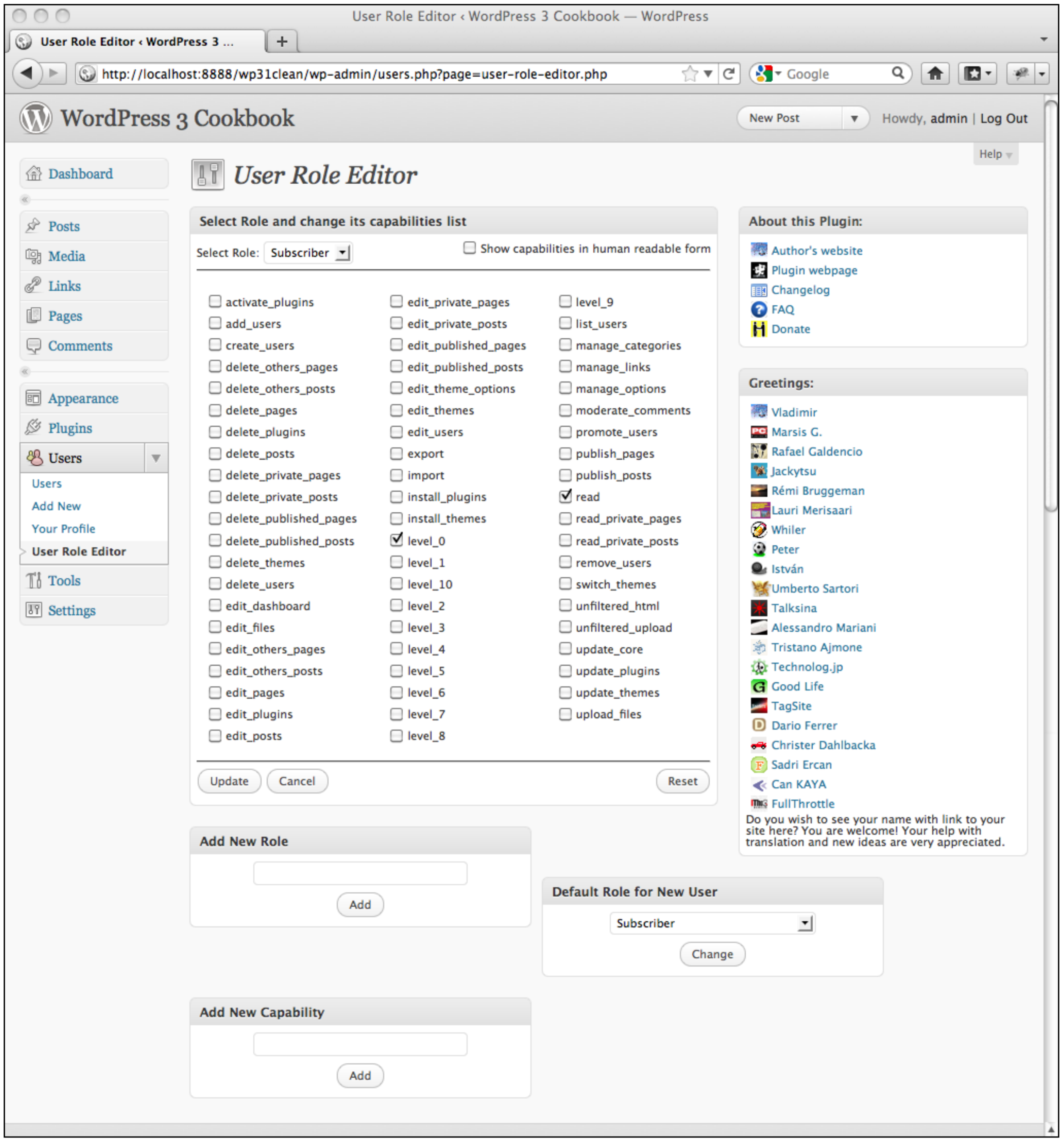
Your changes will now impact all users assigned to the role you have edited.

You cannot modify the Admin role.

To create a new role, follow these steps:

1. Log in to your WordPress dashboard.
2. Click on the **Users** menu.
3. Click on the options **User Role Editor**.
4. On the page that loads, enter a name for the new role in the **Add New Role** field.
5. Click on the **Add** button.
6. On the page that loads, select the privileges you want to role to enjoy.

7. Click on the **Update** button to save your changes.


User Role Editor ‹ WordPress 3 Cookbook — WordPress
User Role Editor ‹ WordPress 3 ...
http://localhost:8888/wp31clean/wp-admin/users.php?page=user-role-editor.php
Google
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Users
Add New
Your Profile
User Role Editor
Tools
Settings
User Role Editor
Select Role and change its capabilities list
About this Plugin:
Author's website
Plugin webpage
Changelog
FAQ
Donate
Greetings:
Vladimir
Marsis G.
Rafael Galdencio
Jackytsu
Rémi Bruggeman
Lauri Merisaari
Whiler
Peter
István
Umberto Sartori
Talksina
Alessandro Mariani
Tristano Ajmone
Technolog.jp
Good Life
TagSite
Dario Ferrer
Christer Dahlbacka
Sadri Ercan
Can KAYA
FullThrottle
Update
Cancel
Reset
Add New Role
Add
Default Role for New User
Change
Add New Capability
Add

Select Role: Subscriber

☐ Show capabilities in human readable form

☐ activate_plugins
☐ add_users
☐ create_users
☐ delete_others_pages
☐ delete_others_posts
☐ delete_pages
☐ delete_plugins
☐ delete_posts
☐ delete_private_pages
☐ delete_private_posts
☐ delete_published_pages
☐ delete_published_posts
☐ delete_themes
☐ delete_users
☐ edit_dashboard
☐ edit_files
☐ edit_others_pages
☐ edit_others_posts
☐ edit_pages
☐ edit_plugins
☐ edit_posts
☐ edit_private_pages
☐ edit_private_posts
☐ edit_published_pages
☐ edit_published_posts
☐ edit_theme_options
☐ edit_themes
☐ edit_users
☐ export
☐ import
☐ install_plugins
☐ install_themes
☑ level_0
☐ level_1
☐ level_10
☐ level_2
☐ level_3
☐ level_4
☐ level_5
☐ level_6
☐ level_7
☐ level_8
☐ level_9
☐ list_users
☐ manage_categories
☐ manage_links
☐ manage_options
☐ moderate_comments
☐ promote_users
☐ publish_pages
☐ publish_posts
☑ read
☐ read_private_pages
☐ read_private_posts
☐ remove_users
☐ switch_themes
☐ unfiltered_html
☐ unfiltered_upload
☐ update_core
☐ update_plugins
☐ update_themes
☐ upload_files

Do you wish to see your name with link to your site here? You are welcome! Your help with translation and new ideas are very appreciated.

Default Role for New User: Subscriber

The fields marked **level_x** are only applicable to older, version 2 installations of WordPress and should not be used for WordPress 3. Note also that you can make the role capabilities easier to read by clicking on the check box labeled **Show capabilities in human readable form**, at the top right of the page.

How it works...

The plugin edits the default role settings and, by interfacing with the database, allows for creation of new roles. Those of you who wish to explore modifying roles without the use of a plugin will need to explore the WordPress Plugin API at `http://codex.wordpress.org/Plugin_API`

See also

- *Managing users* section in this chapter
- *Setting up editorial workflow* section in this chapter

Setting up editorial workflow

If you allow multiple people to post articles and pages to your site, you will want to stay up to date on what your authors and contributors are doing, and you may want to set up a system that allows posts to be reviewed and edited prior to publication. While you can always manage this manually by sending e-mails back and forth, that approach is far from ideal and can be quite a time-consuming task. A better solution to this problem is found in a plugin called Edit Flow.

Edit Flow is a complex plugin. It not only adds content notifications and review process, but also gives the ability to create custom status posts and groups for your users. There are also features appropriate for an online publications, such as an editorial calendar and a story budget feature. If you do not need all the features, the plugin allows you to only enable those things you require.

In this recipe we look how the Edit Flow plugin can be used to create a manageable editorial workflow for content creation on your site.

Getting ready

To execute this recipe, you will need to install the Edit Flow plugin. You will need to install this plugin before you can get started. Search for **Edit Flow** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://editflow.org/`

How to do it...

Let's start out by configuring Edit Flow for basic article submissions and a review process:

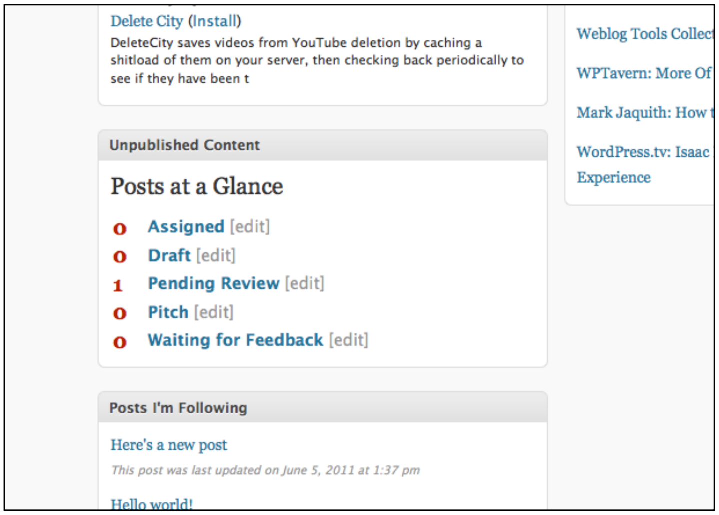
1. Log in to the WordPress **Dashboard**.
2. Click on the new menu named **Edit Flow**.
3. On the configuration screen, de-select **Enable Edit Flow Calendar** and **Enable Story Budget**. Also select the option **Always Notify Admin**.
4. Click on **Save Changes**.

How it works...

You now have a basic editorial review process in place, with notifications being sent to the site admin every time critical actions occur. The plugin has automatically added a set of custom statuses for your posts and pages. Click on the link **Custom Status** to see the list. The plugin has also created new usergroups. View the groups by clicking on the **Usergroups** option in the **Edit Flow** menu.

Taken together, the changes allow a site user (assuming they have permission to create content!) to submit an article for review. Notifications will be sent to higher-level users, who can then log in and comment privately on the content of the posts. When comments are made, the author is notified. This process can be repeated as many times as necessary. Once the post is ready for publication, the status of the post can be changed to published, thereby completing the editorial cycle.

When you log in to the dashboard, a new Edit Flow widget shows you a list of the posts in the editorial process, as shown in the following screenshot:



Delete City (Install)
DeleteCity saves videos from YouTube deletion by caching a shitload of them on your server, then checking back periodically to see if they have been t

Unpublished Content

Posts at a Glance

0 Assigned [edit]
0 Draft [edit]
1 Pending Review [edit]
0 Pitch [edit]
0 Waiting for Feedback [edit]

Posts I'm Following

Here's a new post
This post was last updated on June 5, 2011 at 1:37 pm

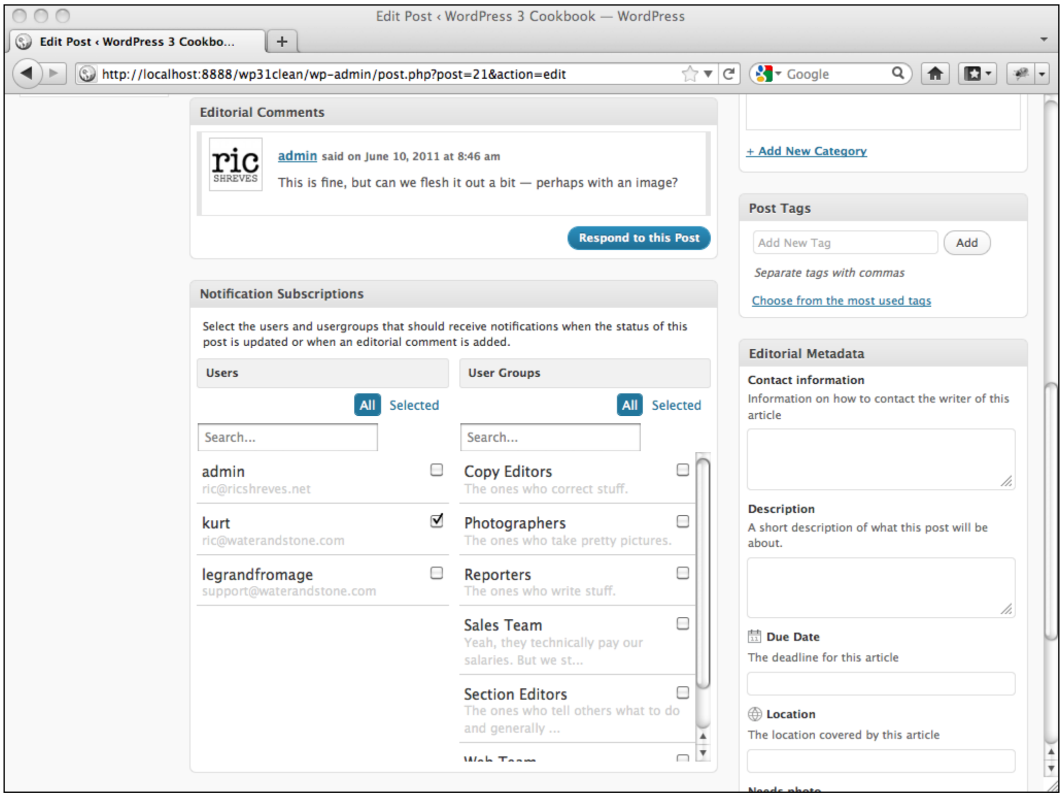
Hello world!

Weblog Tools Collec
WPTavern: More Of
Mark Jaquith: How t
WordPress.tv: Isaac Experience

There's more...

As the next screenshot shows, the editing page for each post now contains several extra field:

- The **Editorial Comments** field is where the editors can comment privately on the post for the author's benefit.
- The **Notifications Subscriptions** section allows you to specify who will receive notifications and can avoid your site editors and admins from being bombarded with unwanted e-mails.
- The **Editorial Metadata** fields give you a way to capture useful information about the post for your internal records. These fields can be customized from the Edit Flow menu.


Edit Post ‹ WordPress 3 Cookbook — WordPress
Edit Post ‹ WordPress 3 Cookbo...
Google
Editorial Comments
ric SHREVES
Respond to this Post
Notification Subscriptions
Users
User Groups
All
Selected
Search...
+ Add New Category
Post Tags
Add New Tag
Add
Choose from the most used tags
Editorial Metadata
Needs photo

http://localhost:8888/wp31clean/wp-admin/post.php?post=21&action=edit

admin said on June 10, 2011 at 8:46 am

This is fine, but can we flesh it out a bit — perhaps with an image?

Select the users and usergroups that should receive notifications when the status of this post is updated or when an editorial comment is added.

Users		User Groups	
admin ric@ricshreves.net	☐	Copy Editors The ones who correct stuff.	☐
kurt ric@waterandstone.com	☑	Photographers The ones who take pretty pictures.	☐
legrandfromage support@waterandstone.com	☐	Reporters The ones who write stuff.	☐
		Sales Team Yeah, they technically pay our salaries. But we st...	☐
		Section Editors The ones who tell others what to do and generally ...	☐
		Web Team	☐

Separate tags with commas

Contact information
Information on how to contact the writer of this article

Description
A short description of what this post will be about.

Due Date
The deadline for this article

Location
The location covered by this article

See also

- *Managing users* section in this chapter
- *Gaining control over user roles and permissions* section in this chapter

Importing and exporting content

WordPress features a very useful script to import your posts, comments, and links from another platform to WordPress. The system also allows you to export your current blog content.

Getting ready

Everything you need to complete this recipe is included inside the WordPress dashboard. Note, however, that while the WordPress import function is included in the dashboard, you will be prompted to download individual importers.

How to do it...

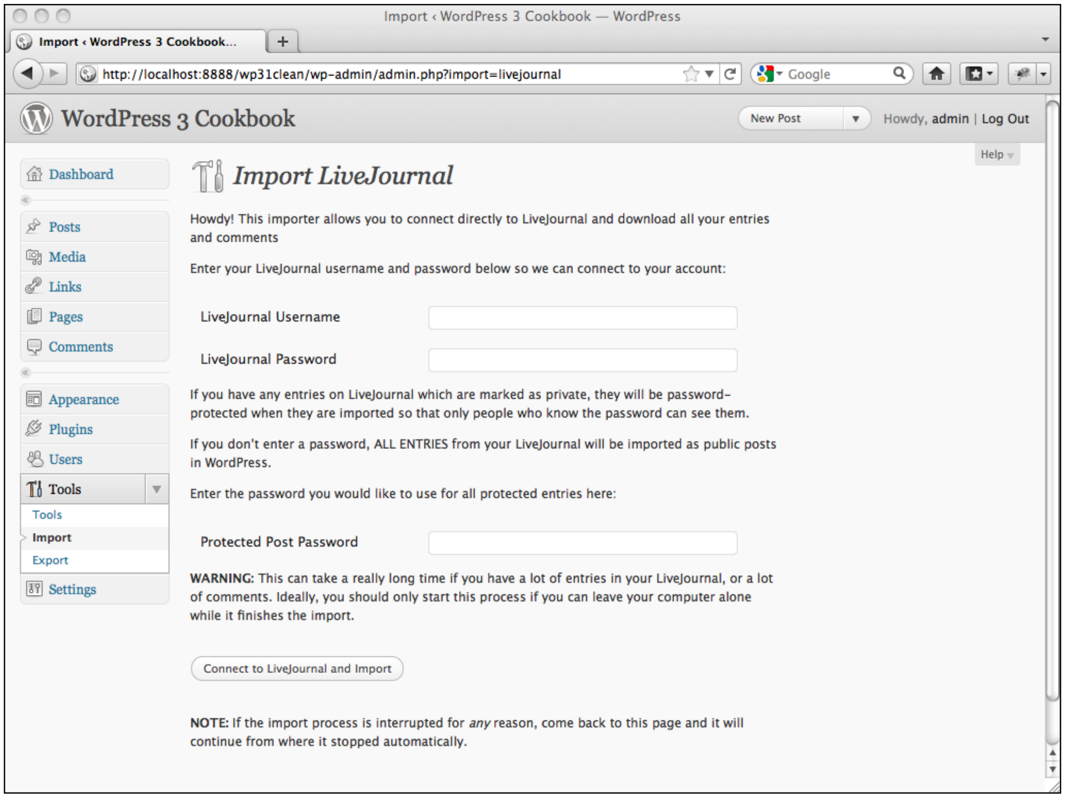
WordPress supports the importation of blog content from several other platforms, including Blogger, Blogroll, LiveJournal, MovableType, and TypePad. It also makes it easier to import posts saved from another WordPress site. The functionality also includes options to help import categories, tags, and RSS feeds.

By way of example, let's assume you wish to import content from a LiveJournal blog site. Follow these steps:

1. Connect to your old blog and export your content. Save the file on your hard drive.
2. Log in to your WordPress **Dashboard**.
3. Click on the **Tools** menu.
4. Click on the option **Import**.
5. On the page that loads, you can select the type of import. In this example, we're going to select the option **LiveJournal**.
6. The system will now prompt you to install the **LiveJournal** import plugin. Install the plugin and activate it.
7. Once done, input your **LiveJournal Username** and **Password** on the fields provided, as shown in the next screenshot.
8. You're done! Please note that depending on your exported file size, this procedure can take a while.

The system will now attempt to import the posts and add them to your WordPress site.

> The process used for a LiveJournal import is typical of that used by all the import options. If you wish to import from a system not listed, check the WordPress plugins listings at `www.WordPress.com`.


Import ‹ WordPress 3 Cookbook — WordPress
Import ‹ WordPress 3 Cookbook...
http://localhost:8888/wp31clean/wp-admin/admin.php?import=livejournal
Google
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Tools
Import
Export
Settings
Connect to LiveJournal and Import

Import LiveJournal

Howdy! This importer allows you to connect directly to LiveJournal and download all your entries and comments

Enter your LiveJournal username and password below so we can connect to your account:

LiveJournal Username

LiveJournal Password

If you have any entries on LiveJournal which are marked as private, they will be password-protected when they are imported so that only people who know the password can see them.

If you don't enter a password, ALL ENTRIES from your LiveJournal will be imported as public posts in WordPress.

Enter the password you would like to use for all protected entries here:

Protected Post Password

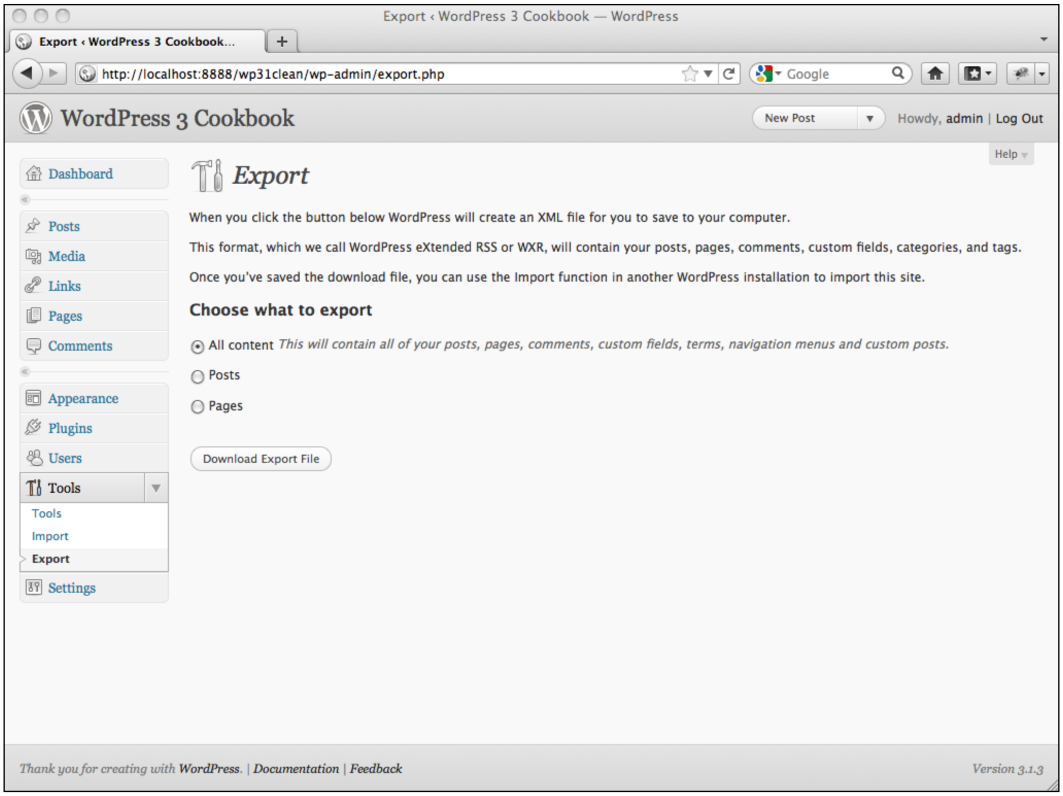
WARNING: This can take a really long time if you have a lot of entries in your LiveJournal, or a lot of comments. Ideally, you should only start this process if you can leave your computer alone while it finishes the import.

NOTE: If the import process is interrupted for *any* reason, come back to this page and it will continue from where it stopped automatically.

Exporting content is also quite easy; simply carry out the following steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Tools** menu.
3. Click on the **Export** option.
4. Select what you wish to export and click on the appropriate radio button, as shown in the next screenshot.
5. Click on the **Download Export File** button.

The system will then prompt you to save the export file.


Export ‹ WordPress 3 Cookbook — WordPress
Export ‹ WordPress 3 Cookbook...
http://localhost:8888/wp31clean/wp-admin/export.php
Google
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Tools
Import
Export
Settings
Download Export File
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

Export

When you click the button below WordPress will create an XML file for you to save to your computer.

This format, which we call WordPress eXtended RSS or WXR, will contain your posts, pages, comments, custom fields, categories, and tags.

Once you've saved the download file, you can use the Import function in another WordPress installation to import this site.

Choose what to export

● All content *This will contain all of your posts, pages, comments, custom fields, terms, navigation menus and custom posts.*

○ Posts

○ Pages

Installing and using Jetpack

In early 2011, Automattic released Jetpack, a cloud-based suite of extensions for the WordPress CMS. Jetpack's features are based in part on functionality users of `www.WordPress.com` have enjoyed for some time. The modules include:

- A site traffic statistics function
- A LaTeX plugin
- A spellchecker
- A Twitter widget
- A shortlinks functionality
- A social sharing mechanism
- Easy embeds from video and media sites

Additional functionality is in the pipeline.

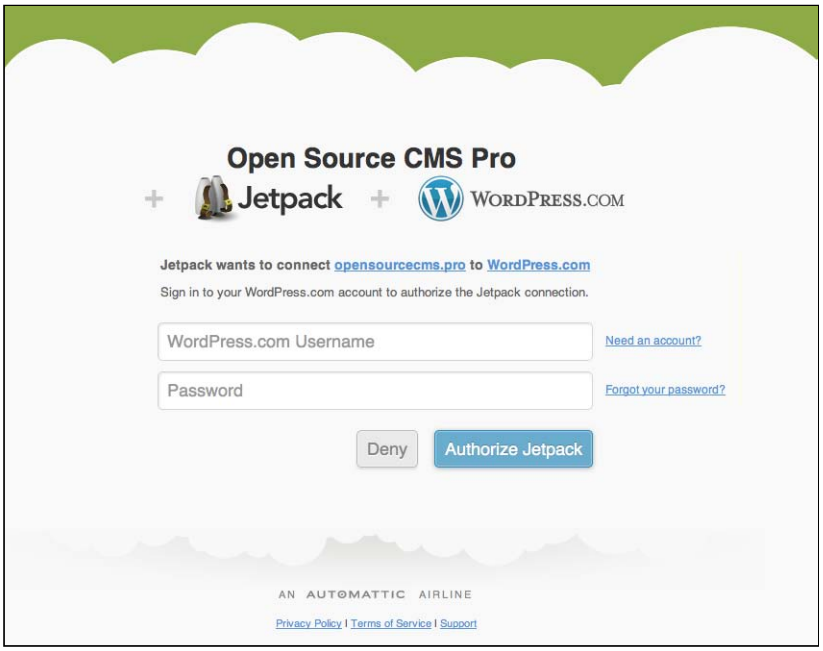
While some of the features of Jetpack are also available in other plugins, Jetpack is a quick and easy solution from a known solutions provider.

In this recipe, we look at installing Jetpack and getting it up and running.

Getting ready

To execute this recipe, you will need to install the Jetpack plugin. You will need to install this plugin before you can get started. Search for **Jetpack** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

Installation requires a slightly different approach than for most other plugins. To use Jetpack, you must be a registered user on `www.WordPress.com`. As the next screenshot shows, you will need to use your `www.WordPress.com` credentials to activate the plugin:



Open Source CMS Pro

Jetpack wants to connect opensourcecms.pro to WordPress.com

Sign in to your WordPress.com account to authorize the Jetpack connection.

+
Jetpack
+
WordPress.com
WordPress.com Username
Need an account?
Password
Forgot your password?
Deny
Authorize Jetpack
AN AUTOMATTIC AIRLINE
Privacy Policy | Terms of Service | Support

How to do it...

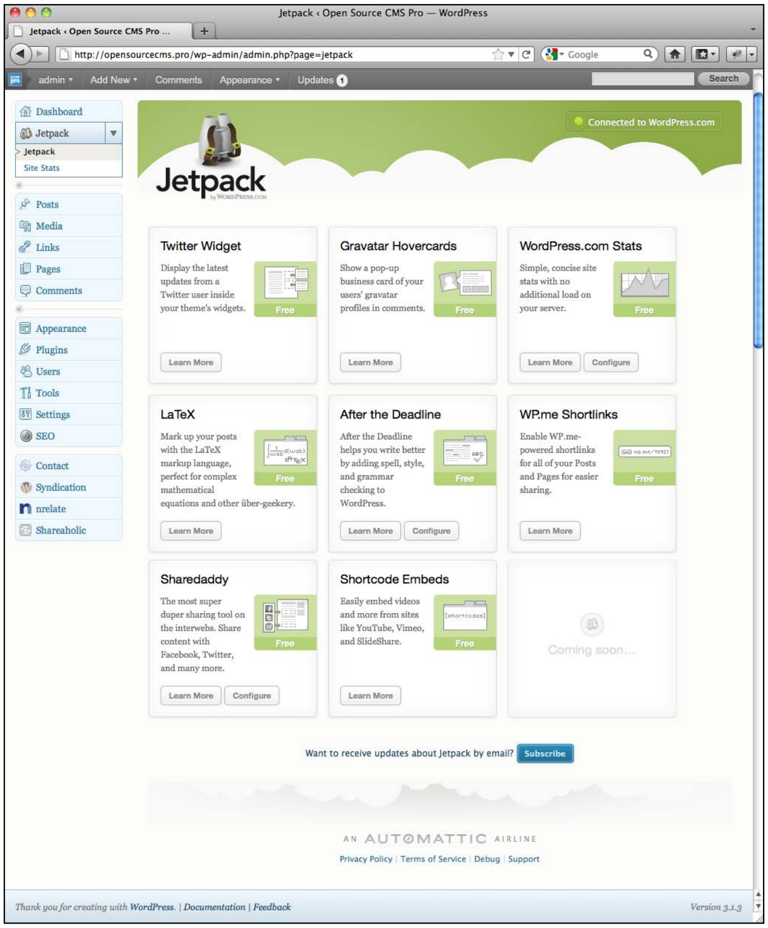
After you have installed Jetpack and authorized it with `www.WordPress.com`, you will see the **Jetpack Dashboard**, as shown in the next screenshot. Before you can start using the features, you will need to do a bit of configuration, as follows:

1. To begin using any of the features, first click on the **Learn More** button.

2. The module will now be activated; if you don't want to use it, click on the **Deactivate** button.
3. Some modules will present you with a **Configure** button, indicating there are customization options available. Click on the **Configure** button to learn what options are available.
4. Some items, such as the **Twitter Widget**, are actually controlled in a separate location.

Once Jetpack is installed, a link to the Jetpack settings page is always visible at the top left of your WordPress dashboard. A separate link will show you the site statistics, assuming you have activated the feature.

> Site statistics and the WP.me shortlinks functionality are also available from the front-end, if you have installed the admin toolbar for the front-end of the site. See the next recipe for a discussion of the admin toolbar.


Jetpack ‹ Open Source CMS Pro — WordPress
Jetpack ‹ Open Source CMS Pro ...
http://opensourcecms.pro/wp-admin/admin.php?page=jetpack
Google
admin
Add New
Comments
Appearance
Updates 1
Search
Dashboard
Jetpack
Jetpack
Site Stats
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
SEO
Contact
Syndication
nrelate
Shareaholic
Connected to WordPress.com
Jetpack
by WordPress.com

Twitter Widget

Display the latest updates from a Twitter user inside your theme's widgets.

Free
Learn More

Gravatar Hovercards

Show a pop-up business card of your users' gravatar profiles in comments.

Free
Learn More

WordPress.com Stats

Simple, concise site stats with no additional load on your server.

Free
Learn More
Configure

LaTeX

Mark up your posts with the LaTeX markup language, perfect for complex mathematical equations and other über-geekery.

Free
Learn More

After the Deadline

After the Deadline helps you write better by adding spell, style, and grammar checking to WordPress.

ABC
Free
Learn More
Configure

WP.me Shortlinks

Enable WP.me-powered shortlinks for all of your Posts and Pages for easier sharing.

wp.me/tk421
Free
Learn More

Sharedaddy

The most super duper sharing tool on the interwebs. Share content with Facebook, Twitter, and many more.

Free
Learn More
Configure

Shortcode Embeds

Easily embed videos and more from sites like YouTube, Vimeo, and SlideShare.

[shortcodes]
Free
Learn More
Coming soon...

Want to receive updates about Jetpack by email?

Subscribe
AN AUTOMATTIC AIRLINE
Privacy Policy | Terms of Service | Debug | Support

Thank you for creating with WordPress. | Documentation | Feedback

Version 3.1.3

How it works...

While some of the features in Jetpack are simple plugins, other features rely on a cloud-based approach to services. Both the stats and the shortlinks functionalities need a connection to the Internet and to `www.WordPress.com`.

There's more...

WordPress Popular Posts is a separate plugin that uses the information gathered in the Jetpack stats module. The plugin provides a nice and configurable widget that shows a list of the most popular posts on your site.

> Learn more about the plugin by visiting the developer's site at `http://polpoinodroidi.com/wordpress-plugins/wordpresscom-popular-posts/`

Enabling the toolbar for users and administrators

One of the new features included with version 3 of WordPress is the **Admin Bar**. The bar is a menu bar that sticks to the top of the page and is always visible. It can be activated for either the front-end, the back-end or both. The bar contains shortcuts to the Admin dashboard and to various administration functions, depending largely on the user role of the viewer.

This recipe shows you how to enable this useful feature for your site users.

Getting ready

Everything you need to complete this recipe is included inside the WordPress dashboard. Note, however, that while the WordPress import function is included in the dashboard, you will be prompted to download individual importers.

How to do it...

This feature is not enabled by default, so you need to set it up for your users. Here's how to do it:

1. Log in to the WordPress **Dashboard**.
2. Click on the **Users** menu.
3. Click on the name of the user for whom you wish to enable the **Admin Bar**.
4. On the user **Profile** page, look for the option labeled **Show Admin Bar**.
5. Select whether you want to user to see it on the front-end, the back-end, or both.
6. Click on **Update Profile**.

2

Installing and Customizing Themes

In this chapter, we will cover:

- How to install a new theme
- Creating a new child theme
- How to modify the colors of a theme
- How to modify the fonts used by a theme
- Creating and integrating a favicon
- Adding a custom logo to a theme
- Customizing the login page
- Using conditional tags to control content display
- Using multiple page templates
- Using post formats
- Creating a custom 404 error page
- Using a static page for your home page
- Adding custom styles to your theme
- Making your site mobile device friendly

Introduction

themes are the most visible portion of your website. The theme controls the appearance of the site, and is largely responsible for the first impression your site makes to visitors.

While the default WordPress installation includes a couple of simple themes, those themes are not right for everyone and are already widely used by others. If you want your site to stand out, you will want to create a distinctive appearance for your site.

Creating the right look for your site can be achieved by customizing an existing theme, adding new theme, or a mix of these approaches. In this chapter, we look at how to work with theme files with the goal of helping you to achieve a unique or distinctive look and feel for your WordPress site.

Installing a theme

A theme is actually a set of files — templates, stylesheets and other helper files — that control the display of your site. There are quite a few themes available on the WordPress.org site and they are all free for your use. Alternatively, there are commercial theme providers and you can always hire something to design a theme for you, or you can even do it yourself. Regardless of how you source your theme, you will need to install it on your site. In this recipe, we look at how to install and activate a new theme.

> To check out the free themes on the official WordPress site, visit `http://wordpress.org/extend/themes/`

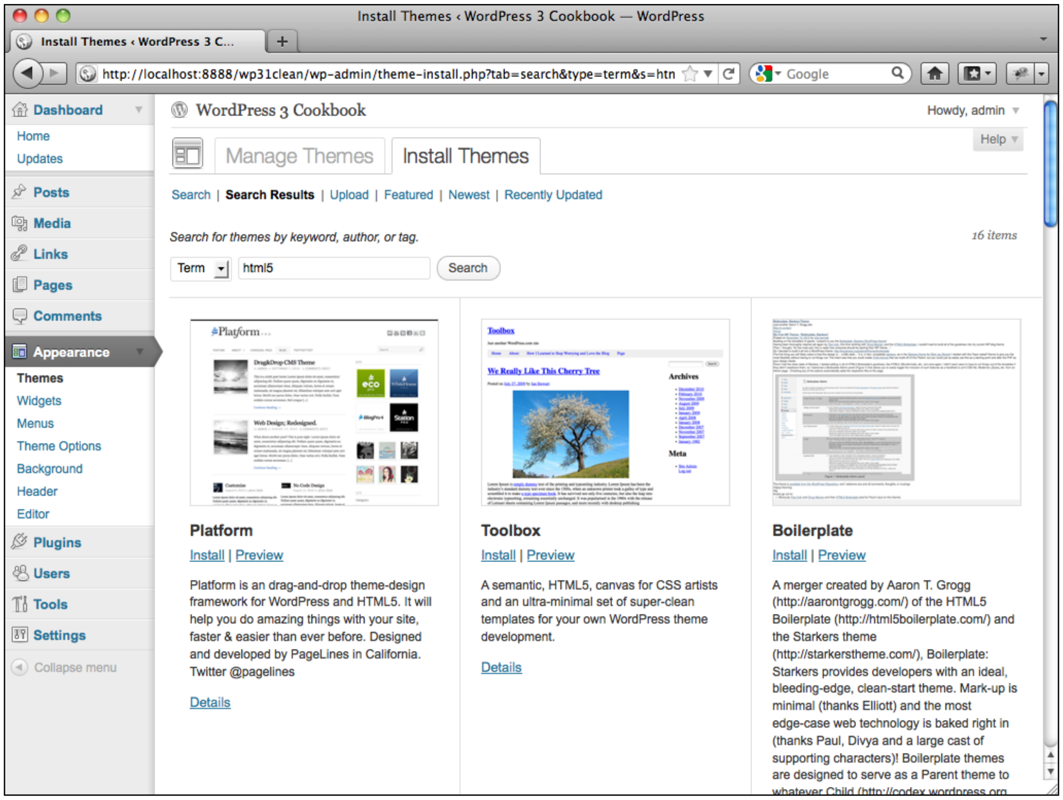
Getting ready

To install theme automatically, you don't need anything special. The entire recipe can be completed from within the WordPress dashboard. If, however, you wish to install the theme manually, as explained later in this recipe, then you will need to have the theme files as well as access to the WordPress installation on your server.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.

3. Click on the **Themes** option.
4. Click on the **Install Themes** tab.
5. In the empty text field at the top of the page, either enter the name of the theme you wish to install, or a search term.
6. Click on the **Search** button.
7. Any themes that match your query will appear, as shown in the following screenshot:


Install Themes ‹ WordPress 3 Cookbook — WordPress
Install Themes ‹ WordPress 3 C...
http://localhost:8888/wp31clean/wp-admin/theme-install.php?tab=search&type=term&s=htn
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Theme Options
Background
Header
Editor
Plugins
Users
Tools
Settings
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
Manage Themes
Install Themes
Search | Search Results | Upload | Featured | Newest | Recently Updated
Search
Platform
Install | Preview
Details
Toolbox
Install | Preview
Details
Boilerplate
Install | Preview

Search for themes by keyword, author, or tag.

16 items

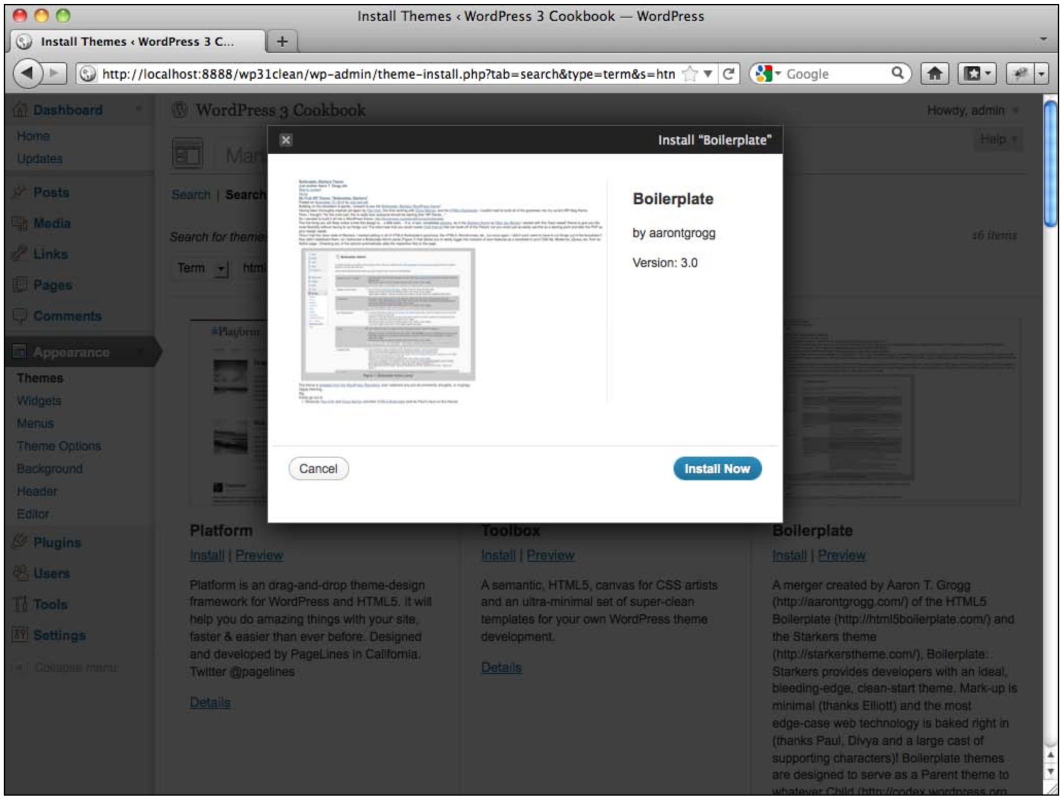
Term: html5

Platform is an drag-and-drop theme-design framework for WordPress and HTML5. It will help you do amazing things with your site, faster & easier than ever before. Designed and developed by PageLines in California. Twitter @pagelines

A semantic, HTML5, canvas for CSS artists and an ultra-minimal set of super-clean templates for your own WordPress theme development.

A merger created by Aaron T. Grogg (http://aarontgrogg.com/) of the HTML5 Boilerplate (http://html5boilerplate.com/) and the Starkers theme (http://starkerstheme.com/), Boilerplate: Starkers provides developers with an ideal, bleeding-edge, clean-start theme. Mark-up is minimal (thanks Elliott) and the most edge-case web technology is baked right in (thanks Paul, Divya and a large cast of supporting characters)! Boilerplate themes are designed to serve as a Parent theme to whatever Child (http://codex.wordpress.org

8. Click on the **Install** link beneath the name of the theme you wish to install. The system will ask you to confirm the installation, as seen in the next screenshot:

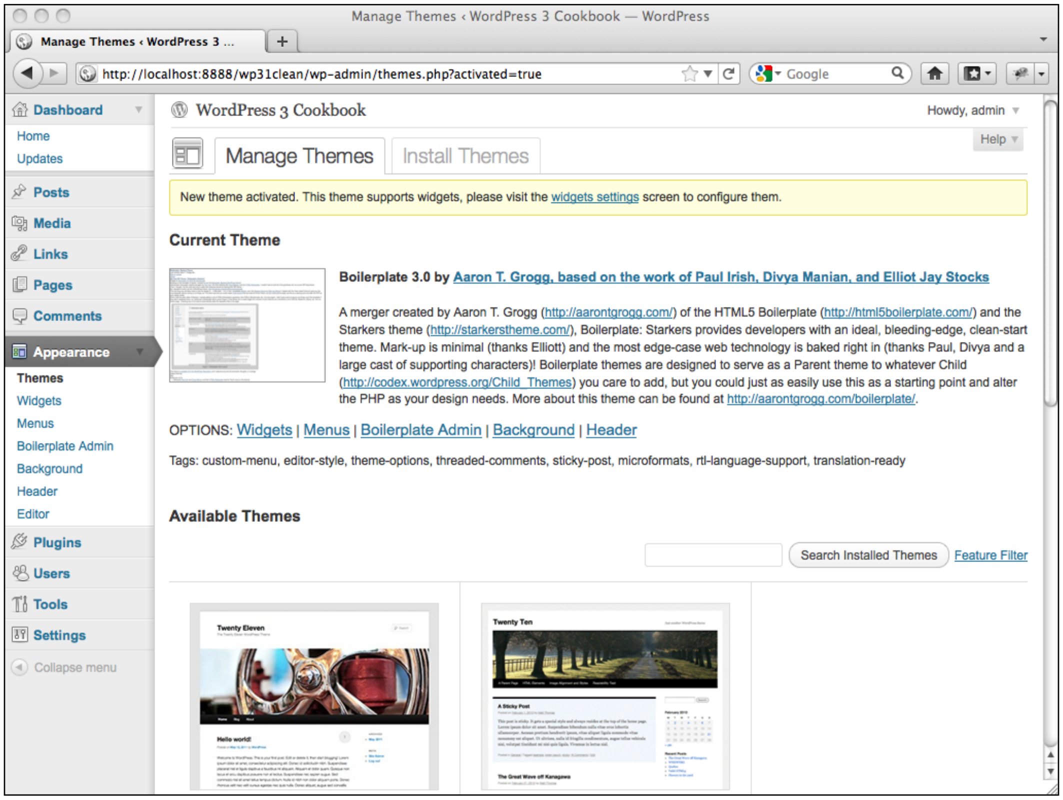

Install Themes ‹ WordPress 3 Cookbook — WordPress
Install Themes ‹ WordPress 3 C...
http://localhost:8888/wp31clean/wp-admin/theme-install.php?tab=search&type=term&s=htm
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Theme Options
Background
Header
Editor
Plugins
Users
Tools
Settings
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
Search | Search
Search for themes
Term
16 items
Install "Boilerplate"
Boilerplate
by aarontgrogg
Version: 3.0
Cancel
Install Now
Platform
Install | Preview
Details
Toolbox
Install | Preview
Details
Boilerplate
Install | Preview

Platform is an drag-and-drop theme-design framework for WordPress and HTML5. It will help you do amazing things with your site, faster & easier than ever before. Designed and developed by PageLines in California. Twitter @pagelines

A semantic, HTML5, canvas for CSS artists and an ultra-minimal set of super-clean templates for your own WordPress theme development.

A merger created by Aaron T. Grogg (http://aarontgrogg.com/) of the HTML5 Boilerplate (http://html5boilerplate.com/) and the Starkers theme (http://starkerstheme.com/), Boilerplate: Starkers provides developers with an ideal, bleeding-edge, clean-start theme. Mark-up is minimal (thanks Elliott) and the most edge-case web technology is baked right in (thanks Paul, Divya and a large cast of supporting characters)! Boilerplate themes are designed to serve as a Parent theme to whatever Child (http://codex.wordpress.org

9. Click on the **Install Now** button.
10. If you are successful, you will see a confirmation message.
11. If you want to use your new theme immediately, click the **Activate** link.
12. The system will then activate the theme and return to the Theme Manager where you will see a confirmation message, as shown in the following screenshot:


Manage Themes ‹ WordPress 3 Cookbook — WordPress
Manage Themes ‹ WordPress 3 ...
http://localhost:8888/wp31clean/wp-admin/themes.php?activated=true
Google
WordPress 3 Cookbook
Howdy, admin
Help
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Boilerplate Admin
Background
Header
Editor
Plugins
Users
Tools
Settings
Collapse menu
Manage Themes
Install Themes
Search Installed Themes
Feature Filter
Twenty Eleven
Twenty Ten

New theme activated. This theme supports widgets, please visit the widgets settings screen to configure them.

Current Theme

Boilerplate 3.0 by Aaron T. Grogg, based on the work of Paul Irish, Divya Manian, and Elliot Jay Stocks

A merger created by Aaron T. Grogg (http://aarontgrogg.com/) of the HTML5 Boilerplate (http://html5boilerplate.com/) and the Starkers theme (http://starkerstheme.com/), Boilerplate: Starkers provides developers with an ideal, bleeding-edge, clean-start theme. Mark-up is minimal (thanks Elliott) and the most edge-case web technology is baked right in (thanks Paul, Divya and a large cast of supporting characters)! Boilerplate themes are designed to serve as a Parent theme to whatever Child (http://codex.wordpress.org/Child_Themes) you care to add, but you could just as easily use this as a starting point and alter the PHP as your design needs. More about this theme can be found at http://aarontgrogg.com/boilerplate/.

OPTIONS: Widgets | Menus | Boilerplate Admin | Background | Header

Tags: custom-menu, editor-style, theme-options, threaded-comments, sticky-post, microformats, rtl-language-support, translation-ready

Available Themes

There's more...

If you are like most users, you will wind up using the automatic installer for your themes. However, if you have built your own theme, or obtained the theme files directly, you may want to use manual installation.

Installing a theme manually

Once you have your theme files, follow these steps:

1. Access your WordPress installation on your server.
2. Navigate to the directory `/wp-content/themes`.
3. Move your theme files into the directory, but located inside their own directory, for example, `/wp-content/themes/mynewtheme`.
4. Log in to your WordPress **Dashboard**.
5. Click on the **Appearance** menu.

6. Click on the **Themes** option.
7. You should see your theme listed on the Theme Manager page. Find it and click on the **Activate** link under the theme's name.

Creating a child theme

In WordPress terminology, a child theme is any theme that inherits the functionality of another theme. The theme that provides the original functionality is referred to as the "parent theme." Child themes are useful, as they allow you to customize the appearance of your site without having to modify the original files, and thereby run into problems when you install upgrades or patches.

Child themes are very easy to create while also giving you a great deal of freedom. Since the attributes of the original theme are inherited, all you really need to put into your child theme are the things you want to modify.

In this recipe, we go through the steps involved in creating a basic child theme and we also look at some of the things you can do with it.

> You can learn more about child themes by visiting the WordPress Codex at `http://codex.wordpress.org/Child_Themes`

Getting ready

For our example, we're going to use the default WordPress Twenty Ten theme. All you need to execute this recipe is your favorite code editing program and access to the WordPress files on your server.

How to do it...

1. Access the WordPress installation on your server.
2. Go the `/wp-content/themes` directory.
3. Create a new directory; name it **twentytwenty**.
4. Create a new blank file named `style.css` and save it to `/wp-content/themes/twentytwenty`.
5. Paste into your new `style.css` file the following code:

```
/*
Theme Name: Twenty Twenty
URI: http://www.yoursite.com/
Description: Child theme based on the Twenty Ten theme
```

```
Author: Author's Name
Author URI: http: //www.authorsurl.com/
Template: twentyten
Version: 1.0
*/
@import url('../twentyten/style.css');
```

6. Save the file.
7. Log in to your WordPress dashboard.
8. Click on the **Appearance** menu.
9. Click on the **Themes** option.
10. You will see on the **Themes** page a list of the themes installed on your site, including your new theme **Twenty Twenty**.
11. **Activate** the Twenty Twenty theme.

When you visit the front-end of your website, you will see no differences. That is because while you have created a new theme, at this point there are no unique styles or functions associated with Twenty Twenty, hence the output is straight from the original TwentyTen.

How it works...

All you need to create a new child theme is the `style.css` file. That file, however, is required and must be named `style.css`. Moreover, you must make several of the declarations you see in the code example above. You must give your child theme a unique **theme name** and you must declare the parent theme in the **template** field. All other values in the code above are optional.

The theme system in WordPress works on a system of inheritance. When you declare the parent theme in the **template** field of your `style.css` file, WordPress will automatically treat the new theme as a child theme.

Note that the child theme's `style.css` file completely replaces the parent theme's stylesheet of the same name. Accordingly, we import the parent theme's stylesheet with the declaration `@import url('../twentyten/style.css');`

> If you only import the parent theme's styles, your theme might be impacted if the parent theme's stylesheet is updated, say, as the result of a theme upgrade. To avoid this problem, you can simply copy all the parent theme's styles into the child theme's stylesheet.

There's more...

The child theme we created above looks identical to the original parent theme. Here's how to customize the output of your new Twenty Twenty to fit your needs.

Customizing the styling of your child theme

To customize the styling of your new child theme, simply place selectors in the new `style.css` file. You can add new selectors, or you can change the definition of any of the original selectors by placing new versions in your stylesheet.

You can also impact the styling through the use of unique templates in your child theme. WordPress will look first for templates in your child theme, then it will look in the parent theme. This means that not only can you add new templates to your child theme, you can override the original templates by creating new ones inside your child theme and giving them the same name as templates in your parent theme.

Customizing the functionality of your child theme

To add your own functions to your child theme, simply create a new file named `functions.php` and place it in the child theme directory. Note that, unlike templates and stylesheets, the `functions.php` file in your child theme will not override the `functions.php` file in the parent theme. This means that the functions of the parent theme will always be used unless you disable those functions. You can simply comment out the original function and add a new, properly named function to your child theme's `functions.php` file.

> Remember, if the parent theme's files are updated and the original `functions.php` file is overwritten, you will have to go back and comment out the relevant functions again!

Modifying your theme colors

Have you ever come across a WordPress theme available online and thought, 'Wow, this is a great theme, but it would look even better if it had a green layout!'? Luckily, changing a theme's color scheme isn't as difficult as it might seem.

In this recipe we look at editing your theme's stylesheets to modify the theme color scheme.

Getting ready

We're going to be working on the files inside your theme. To complete this recipe, you will need an editor to work with the CSS files of your theme; a standard text editor will do the job. You will also need access to the files of your WordPress installation on your server.

Before you get started with making changes, it's a good idea to first sort out the color scheme you want to implement and grab the color codes. One of the best ways to do this is to use a graphics program to work up a mock up of what you want your page to look like. Once you have adjusted things and got them just the way you want them, copy the color codes you need and keep them handy for the steps below.

How to do it...

1. The first thing we need to know is which hexadecimal color codes are currently used in the theme you want to modify. Most WordPress themes use a color scheme of three to five different colors. In order to know which colors are used in the theme, access your theme files and open the CSS file `style.css`.
2. The CSS property used to define a background is called `background-color` (or simply as part of the `background` selector). For the foreground color, the property's name is `color`. For the border colors, it is `border-color` (or simply, as part of the `border` selector). For example, here's the color scheme that's used on the OpenBook theme:
    - Background color: `#151515`
    - Content background color: `#fff`
    - Header blocks: `#222`
    - Green (used for links): `#49AB0D`
    - Blue (Titles, hover links): `#109dd0`
3. Search for the color codes used in the current theme and then replace them with the color codes you saved earlier.
4. Repeat the search and replace the command as necessary.
5. Save the file.

If you now view the front-end of your WordPress site, you should see your new colors in place.

> If you're new to CSS a good starting point is the tutorial at W3C Schools: `http://www.w3schools.com/css/css_intro.asp`

How it works...

CSS, or Cascading Style Sheets, contain the definitions of the various styles used in your theme. By changing the color codes in the stylesheet, we change the colors displayed by the browser when it interprets your theme files.

There's more...

The following are a few important points, for your information:

- All themes use a `style.css` file, but some themes also use additional stylesheets (for example, using a specific stylesheet for Internet Explorer is very common). Thus, make sure to replace colors in all stylesheets that are part of the theme.
- This trick can only replace CSS-based colors. To modify image colors, you'll need to use a design program such as Adobe Photoshop or The Gimp.
- If you changed your theme colors and some parts still display the old colors, make sure that the CSS colors are written in hexadecimal codes (for example: `#151515`). Some theme designers use color names instead of hexadecimal codes (for example: they may use background color: white instead of background color`: #ffffff`).
- Some color codes can be written by using shorthand, for example, `#006699` can be written as `#069`—therefore, make sure that you've checked for that too.

Modifying your theme fonts

Now that you have learned how to search and replace hexadecimal color codes, let's customize your theme a bit more.

In this recipe, we're going to see how we can easily modify the fonts used in a WordPress theme and also discuss best practices for typography in WordPress.

Getting ready

For this recipe, you'll need exactly the same things that were needed in the *Modifying your theme colors* recipe—a theme to customize, and a text editor. As we recommended in the previous recipe, it's a good idea to sort out the font scheme you want to implement before you get started. A good graphic design program will allow you to experiment with different combinations until you find the font scheme you prefer. Once you have it sorted out, note the names of all the fonts used, as you will need that information for the steps that follow.

Web safe fonts

A common beginner's mistake is to try and use non web-safe fonts for a web site. For example, there are web sites using the Myriad Pro or Segoe UI fonts. While those fonts may look great, they are only available on about 10% percent of the browsers that are likely to be used by your readers.

The following fonts are widely supported, even by most older browsers:

- Times New Roman
- Arial
- Verdana
- Courier
- Comic Sans
- Trebuchet MS
- Century Gothic
- Helvetica

You can find a full listing of web safe font combinations at `http://www.w3schools.com/cssref/css_websafe_fonts.asp`.

The list of fonts above is pretty limiting; in the latter part of this recipe we look at a new alternative that has arisen with CSS3.

How to do it...

1. In order to modify the fonts of a theme, the first step is to identify all the font declarations in the theme and find out which fonts are being used presently. Open your theme's `style.css` file and search to find the font and font-family CSS properties.
2. Modify the font declarations you wish to change, adding or substituting the new fonts you wish to see used.
3. Save the `style.css` file.
4. Open the other style sheets in your theme and repeat the process.

Check the front-end of your site and you should see your new themes at work.

There's more...

Here are some useful tips about fonts in general and a way to sidestep the web safe fonts issue.

Tips and things to know about fonts

A few things to keep in mind:

- The optimal font size for text readability is between 11 and 14 pixels; 12px is the most common choice of font size. For titles, (`h1`, `h2`, `h3`, and so on) a font size between 14 and 26 pixels tends to work well.
- While using a font with a two word name (for example, `Trebuchet MS`), always put it between quotes; that is, `font-family: "Trebuchet MS"`;)
- While defining font families always list at least two fonts, and offer a generic family name as the last alternative. The client's browser will use the first font that it recognizes.

Using @font-face

CSS3 is the latest standard in stylesheets. It adds a number of interesting properties, including one that has been seen on and off in previous versions of the standard: `@font-face`. As the rule has appeared before, browser support for it is quite good, including with traditionally problematic Internet Explorer. `@font-face` lets you break away from worrying whether a particular font is installed on the user's machine and thereby gives you a wide range of options for your typography.

It works like this: Use of an `@font-face` declaration in your theme will allow you to place your font file in a web directory and then link to it from the theme's CSS. Since the fonts are already hosted on the web, you don't have to rely on the local browser to supply them.

Here's how you can use it:

1. Obtain the font files you need for your site.
2. Move them on to a server, into a directory that is publicly accessible. If this is only for the one site you are working on, you can even place them inside your theme directory.

   ```
   Open up the primary CSS file for your theme and add font
   declaration for your new font. The declaration will look something
   like this:
   @font-face {
      font-family: 'NameOfTheFont';
      src:   url('path-to-my-webfont.eot') format ('eot'),
            url('path-to-my-webfont.woff') format ('woff'),
            url('path-to-my-webfont.ttf') format ('truetype');
   }
   ```

3. Your are ready now to use the new font family in your selectors. Simply specify the `'NameOfTheFont'` as part of a font-family definition, for example:

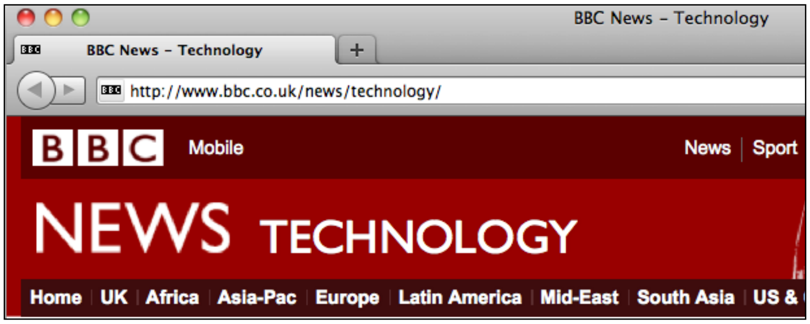
```
body {
   font-family: NameOfTheFont, times, serif;
}
```

> Not all fonts are free for use without restriction. Before you adopt a font for this usage, make sure you have the rights to use it on your site.

Creating and integrating a favicon

A favicon is a small icon (16 x16 pixel) associated with a website. The favicon is displayed by modern web browsers in the address bar, tabs, and bookmarks.

Nowadays, almost all the websites and blogs have their own favicon. The following screenshot shows the BBC website's favicon, as displayed in the Mozilla Firefox browser:


BBC News – Technology
BBC News – Technology
+
http://www.bbc.co.uk/news/technology/
BBC
Mobile
News
Sport
NEWS TECHNOLOGY
Home
UK
Africa
Asia-Pac
Europe
Latin America
Mid-East
South Asia
US &

In this recipe, we look at how you can add a favicon to your site.

Getting ready

To execute this recipe, you will need access to your WordPress installation on your server. You will also need a 16 x 16 pixels image to serve as a favicon. Due to the very small display size of the favicon, the image must be very simple.

> Try using a background color that fits your website's color scheme, together with a simplified version of your logo.

While you can use `.jpg`, `.png`, `.gif`, or even `.mg` and a `.png` file to display a favicon, unfortunately, Internet Explorer (6 and 7) only recognizes the Windows `.ico` file format. Therefore, if you want to have an IE-compatible favicon, you'll have to convert your image file from `.png`, `.gif`, or `.jpg` to Windows `.ico`.

> Many imaging software applications can convert an image into a Windows icon file. There is even an online service called ConvertIcon at `http://converticon.com/`. The ConvertIcon application will also resize your image if needed. Therefore, there's no need to worry about your image width and height.

How to do it...

1. Once you have your favicon ready, upload it to a directory inside the WordPress installation on your server; your theme's directory is a good location, but it is up to you.
2. Open the `header.php` file from your theme.
3. Add the following line of code to the file. This line can be placed anywhere within the `<head>` and `</head>` tags:

   ```
   <link rel="shortcut icon" type="image/x-icon" href="/path/to/your/
   favicon.ico" />
   ```

4. Once you have saved your `header.php` file, your favicon will be displayed.

How it works...

With the addition of the code above, the browser will automatically detect and display the favicon when someone visits the site.

There's more...

If you uploaded your favicon into your `wp-content/themes/yourtheme` directory, we can also use the `bloginfo()` function in order to automatically retrieve the template path as follows:

```
<link rel="shortcut icon" type="image/x-icon"
    href="<?php bloginfo('template_url'); ?>/favicon.ico" />
```

If you chose to use a file format such as `.gif`, `.png`, or `.jpg` for your favicon, you can add your favicon with the following code; however, Internet Explorer may not recognize it:

```
<link rel="icon" type="image/png"
    href="favicon.png" />
```

Adding a custom logo

By default, most WordPress themes display a header text—usually the name of the blog and blog description. This is a nice option for personal blogs. However, I personally believe that displaying your own personal logo will make your blog look even more professional.

In this recipe, we shall learn how we can add a logo instead of the default blog name and slogan on a WordPress theme. The screenshot at the end of the recipe shows the logo integration on a default WordPress theme.

Getting ready

For this recipe, you'll need your own logo and a WordPress theme on which you'd like to integrate your logo. I shall be using the WordPress default theme for this recipe.

However, there are a few things to be kept in mind before getting on with this recipe:

- Some recent themes don't display the blog name and slogan anymore—instead, they display a default logo which can be changed by editing the code or even by defining a new logo in a custom WordPress control panel (which shall be covered in *Chapter 3*).
- Due to the fact that each theme is coded differently, the result of this recipe may vary from one theme to another.

How to do it...

1. Access the directory containing the theme files for your WordPress installation.
2. Copy your logo into the `/images` directory of your active theme; note the name.
3. Open the active theme's `header.php` file.
4. Locate the part of code where the blog name and description are displayed. In the WordPress default theme, in the default Twenty Eleven theme, it looks like this:

```
<hgroup>
   <h1 id="site-title"><span><a href="<?php echo esc_url( home_
url(
      '/' ) ); ?>" title="<?php echo esc_attr( get_bloginfo(
'name',
        'display' ) ); ?>" rel="home"><?php bloginfo( 'name' );
           ?></a></span></h1>
   <h2 id="site-description"><?php bloginfo( 'description' ); ?></
h2>
</hgroup>
```

5. While we could just put an `html` image tag between the `<h1>` and `</h1>` tags, there's a much better, SEO friendly, way to display our logo—by using CSS.

6. Open the `style.css` file from your theme.
7. Find the definition for `#site-title`. In the default Twenty Eleven theme, it looks like this:

```
#site-title {
   margin-right: 270px;
   padding: 3.65625em 0 0;
}
```

8. Now, replace the preceding piece of code with the following:

```
#site-title {
   padding: 3em 0;
   background: url("images/logo.png") no-repeat scroll 0 1em
      transparent;
   margin-right: 270px;}
```

9. Find the definition for `#site-title a`. In the default Twenty Eleven theme, it looks like this:

```
#site-title a {
   color: #111;
   font-size: 30px;
   font-weight: bold;
   line-height: 36px;
   text-decoration: none;
}
```

10. Now, replace the preceding piece of code with the following:

```
#site-title a {
   color: #111;
   font-size: 30px;
   font-weight: bold;
   line-height: 36px;
   text-decoration: none;
   margin-left: 120px;
}
```

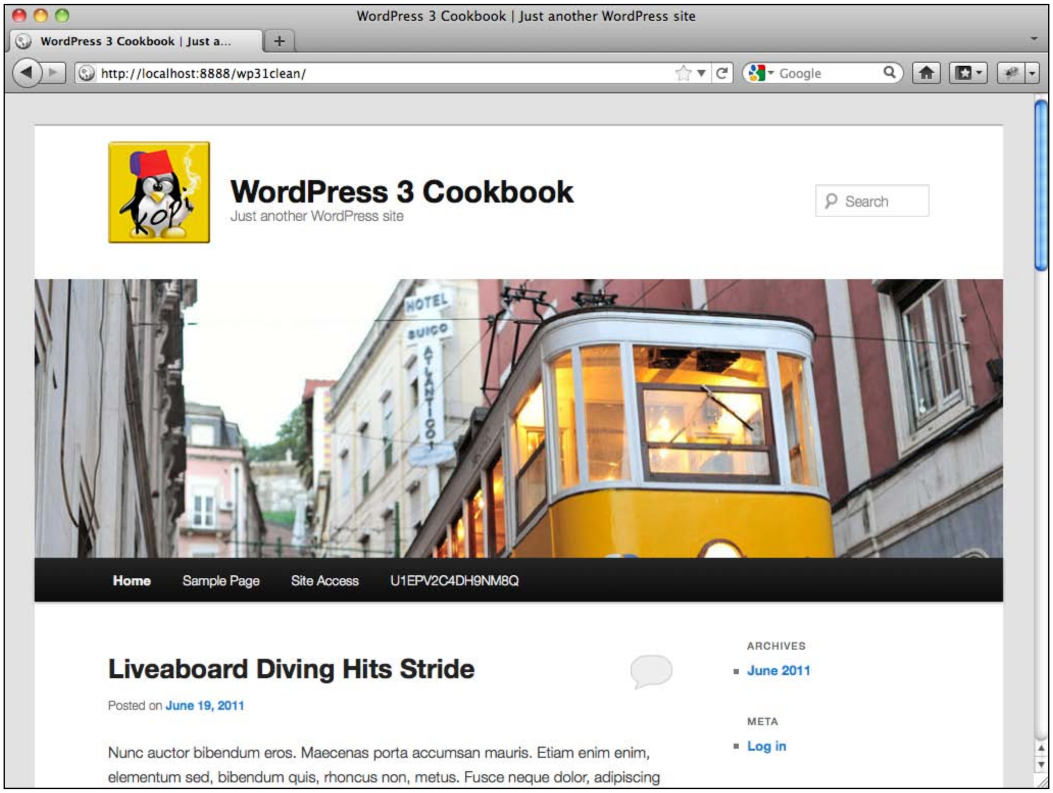
11. Find the definition for `#site-description`. In the default Twenty Eleven theme, it looks like this:

```
#site-description {
   color: #7a7a7a;
   font-size: 14px;
   margin: 0 270px 3.65625em 0;
}
```

12. Now, replace the preceding piece of code with the following:

```
#site-description {
   color: #7a7a7a;
   font-size: 14px;
   margin: -50px 270px 3.65625em 120px;
}
```

If you visit the front-end of the site now and refresh the page, you should see your new logo on the header of your site. The original Twenty Ten theme does not display a logo; you can see in the screenshot below the revised theme with our logo to the left of the title and tagline:


WordPress 3 Cookbook | Just another WordPress site
WordPress 3 Cookbook | Just a...
http://localhost:8888/wp31clean/
Google
WordPress 3 Cookbook
Just another WordPress site
Search
HOTEL
BUICO
ATLANTICO
Home
Sample Page
Site Access
U1EPV2C4DH9NM8Q
ARCHIVES
June 2011
META
Log in

Liveaboard Diving Hits Stride

Posted on June 19, 2011

Nunc auctor bibendum eros. Maecenas porta accumsan mauris. Etiam enim enim, elementum sed, bibendum quis, rhoncus non, metus. Fusce neque dolor, adipiscing

How it works...

The CSS approach to displaying a logo works by using the logo image as a background for the `<h1>` tag. In the first selector, `#site-title`, you can see the background property being used to display the image file. The margin and padding have been adjusted to aid in proper display inside the header. In the `#site-title a` selector, the margin has been adjusted to place the text for the site name to the right side of the logo. Finally, the `#site-description` selector was adjusted to add a margin to place the site description underneath the site title and to the right of the logo image.

Customizing the login page

By default the WordPress login page is outside the theme system and therefore, the login page does not change when the theme changes. By default, all WordPress login pages look the same, with the basic WordPress branding in place and no information that identifies the site.

For many clients, the default presentation is not the best choice. Some prefer to brand the login page, or at least to make it visually consistent with the active theme. In this recipe we look at a plugin that turns the default login page into a page that can be managed through the dashboard and themed via the theme system.

Getting ready

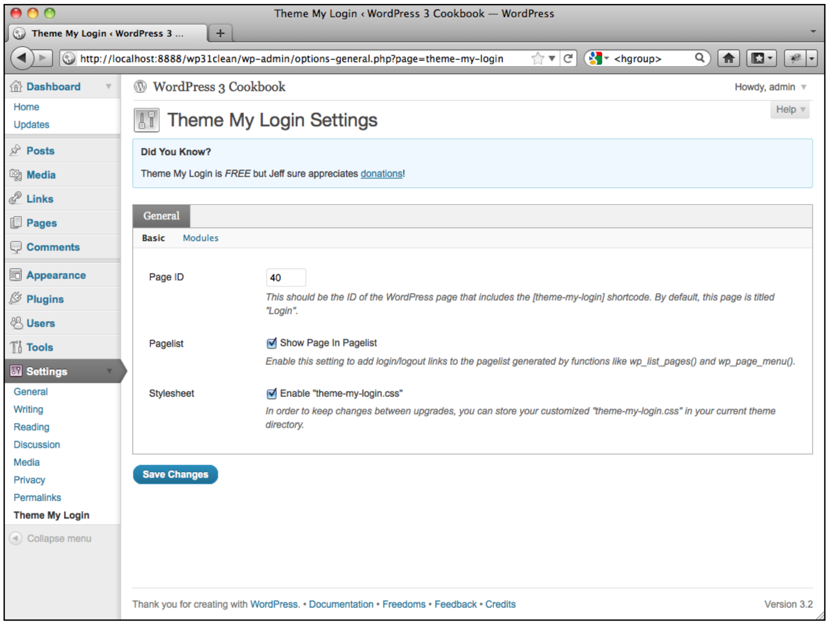
To execute this recipe, you will need to install the Theme My Login plugin. You will need to install this plugin before you can get started. Search for **Theme My Login** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> The developer maintains a basic set of online documentation for the plugin at `http://www.jfarthing.com/docs/Theme_My_Login`

How to do it...

Once the plugin is installed and activated, carry out the following steps to configure the plugin:

1. Log in to the WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the **Theme My Login** option.
4. On the **Basic** tab, the default settings are fine for most people. Simply click on the **Save Changes** button to finalize the configuration.


Theme My Login ‹ WordPress 3 Cookbook — WordPress
Theme My Login ‹ WordPress 3 ...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=theme-my-login
<hgroup>
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Theme My Login
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
General
Basic
Modules
Save Changes
Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits
Version 3.2

Theme My Login Settings

Did You Know?

Theme My Login is *FREE* but Jeff sure appreciates donations!

Page ID: 40

This should be the ID of the WordPress page that includes the [theme-my-login] shortcode. By default, this page is titled "Login".

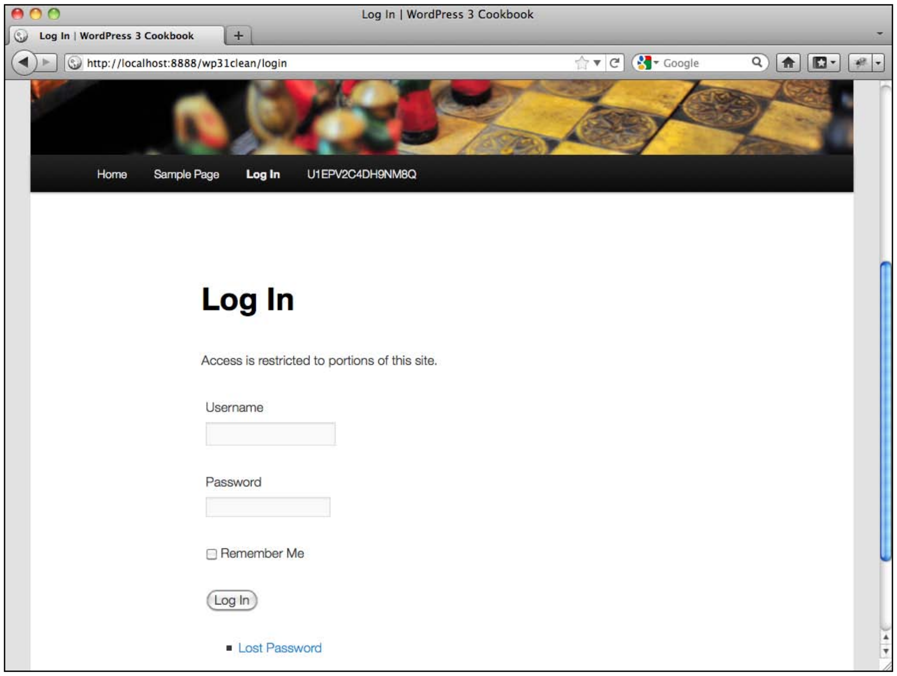
Pagelist: ☑ Show Page In Pagelist

Enable this setting to add login/logout links to the pagelist generated by functions like wp_list_pages() and wp_page_menu().

Stylesheet: ☑ Enable "theme-my-login.css"

In order to keep changes between upgrades, you can store your customized "theme-my-login.css" in your current theme directory.

The new login page is now ready to go. It will be active on your site as soon as you finish the configuration steps, above. The default output is shown in the next screenshot:


Log In | WordPress 3 Cookbook
Log In | WordPress 3 Cookbook
http://localhost:8888/wp31clean/login
Google
Home
Sample Page
Log In
U1EPV2C4DH9NM8Q
Log In
Lost Password

Log In

Access is restricted to portions of this site.

Username

Password

☐ Remember Me

To customize the page, you have several options. You can add text to the page by editing the new login page under the Page Manager. To add text to the page, follow these steps:

1. Log in to the WordPress **Dashboard**.
2. Click on the **Pages** menu.
3. Select the **Log In** page from the list of pages and click on the **Edit** link to open it for editing.
4. Make your changes to the page.
5. Click on the **Update** button to make your changes live.

> When editing the page, you will see the following: `[theme-my-login]`. This is the shortcode that is used to put the login output on the page. You can use that shortcode elsewhere if you prefer, and unpublish the default page created by the plugin.

Another option for customizing the page is to modify the CSS. The plugin adds a new stylesheet to your site just for this purpose, but to use it, you need to take some steps:

1. Access your WordPress installation on the server.
2. Go to the directory `/plugins/theme-my-login`.
3. Copy the file `theme-my-login.css`.
4. Paste the file inside your active theme's directory.

The new `.css` will now control the styling of the form. You can edit it like any other CSS, using either the system's built-in editor, or editing with the application of your choice.

Yet another option for customizing the page is to use a custom template. The plugin adds a number of new stylesheets to your site just for this purpose, but to use them, you need to take the following steps:

1. Access your WordPress installation on the server.
2. Go to the directory `/plugins/theme-my-login`
3. Copy any of the templates you wish to use.
4. Paste the files inside your active theme's directory.

How it works...

The plugin enables a shortcode that allows for the login and related functionality to be displayed in a page or a post. The plugin also comes with a dedicated stylesheet and a set of templates that can be used to control not only the login page, but also all related pages, for example, the register form or the reset password form.

There's more...

The plugin includes several options that extend the login functionality and allow for further customization.

Enabling the plugin's modules

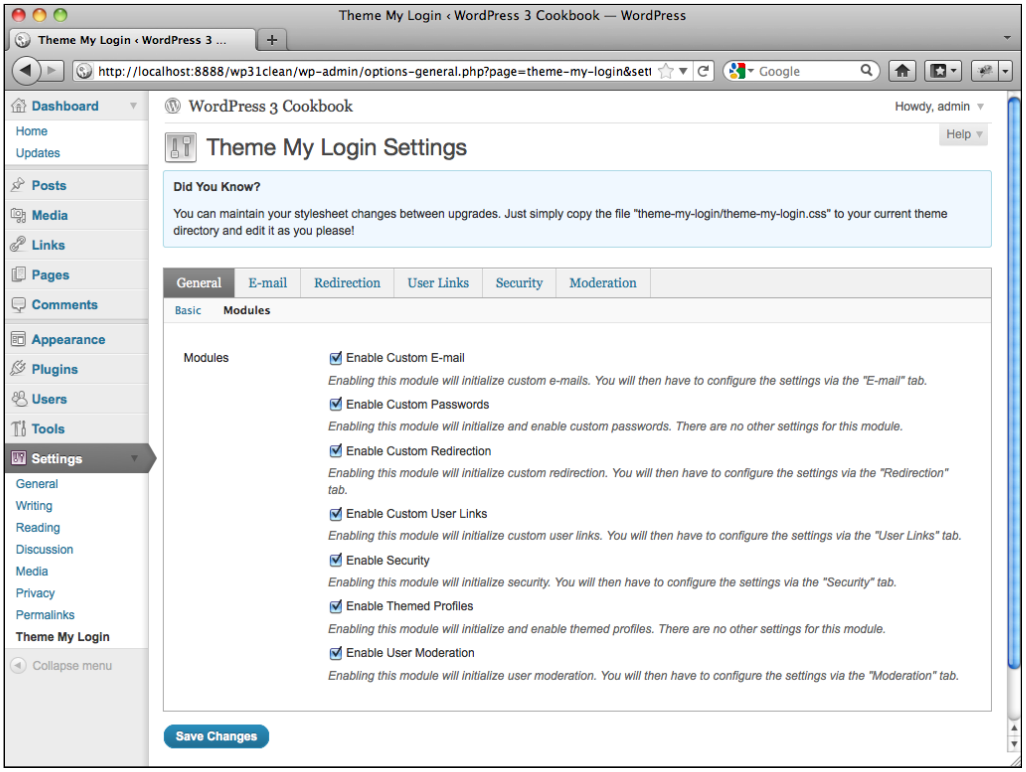
The plugin is bundled with a set of modules that allow you to do any of the following:

- Customize notification e-mails related to the login and registration process
- Enable custom passwords, allowing people to login with their username or e-mail address
- Redirect users to specific pages after submitting registration and login forms
- Add to or edit the links shown to logged in users
- Enhance security by limiting the number of login attempts
- Provide themes profiles for your users
- Enable moderation for particular actions, related primarily to registration

To enable and customize some or all of these functions, follow these steps:

1. Log in to the WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the **Theme My Login** option.
4. Click on the **Modules** tab.
5. Select any of the options you desire.
6. Click on the **Save Changes** button.

7. You will now see a new set of tabs, as shown in the screenshot below. The number of tabs will depend upon which options you have selected. Click on the tabs to configure the options.


Theme My Login ‹ WordPress 3 Cookbook — WordPress
Theme My Login ‹ WordPress 3 ...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=theme-my-login&sett
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Theme My Login
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
General
E-mail
Redirection
User Links
Security
Moderation
Basic
Modules
Save Changes

Theme My Login Settings

Did You Know?

You can maintain your stylesheet changes between upgrades. Just simply copy the file "theme-my-login/theme-my-login.css" to your current theme directory and edit it as you please!

Modules

☑ Enable Custom E-mail

Enabling this module will initialize custom e-mails. You will then have to configure the settings via the "E-mail" tab.

☑ Enable Custom Passwords

Enabling this module will initialize and enable custom passwords. There are no other settings for this module.

☑ Enable Custom Redirection

Enabling this module will initialize custom redirection. You will then have to configure the settings via the "Redirection" tab.

☑ Enable Custom User Links

Enabling this module will initialize custom user links. You will then have to configure the settings via the "User Links" tab.

☑ Enable Security

Enabling this module will initialize security. You will then have to configure the settings via the "Security" tab.

☑ Enable Themed Profiles

Enabling this module will initialize and enable themed profiles. There are no other settings for this module.

☑ Enable User Moderation

Enabling this module will initialize user moderation. You will then have to configure the settings via the "Moderation" tab.

8. Click on the **Save Changes** button when you are finished.

Using conditional tags to control content display

Sometimes, you may want to display some content only on specific pages or sections. For example, you may wish to display a welcome message on your blog homepage or show specific information on the Categories page. In this recipe, we'll learn how to control output with the use of WordPress conditional tags.

Getting ready

To achieve this recipe, you'll need a code editor and access to the files of your WordPress installation.

How to do it...

1. Access your WordPress installation on the server.
2. Navigate to your active theme directory and find the template file you wish to modify.
3. Open the file for editing with your favorite editor.
4. WordPress conditional tags are Boolean variables so you have to use them as a condition of a php `if` statement, as shown in the following example:

```
<?php if(is_page())
{
    echo "Page title:";
    the_title;
} ?>
```

5. The preceding `if` statement will return `false` if the current page template isn't a WordPress page template so nothing would happen. Otherwise, the `if` statement will return `true` and will print the page title on screen.
6. After you've added the logic to the template you desire, save the file.

How it works...

The PHP conditional statement frames the tag and tells the system to check and see if the condition is true. Essentially, it works like this:

```
<?php if (conditional_tag()) "
{
    //Do something only if the condition is met.
} ?>
```

On the other hand, some conditional tags request a parameter to work. For example, to use the `is_year()` tag, you have to provide a year in the parameter.

```
<?php if (is_year("2007"))
{
    //Do something only if the post or page was published
      during the year 2007.
} ?>
```

Most conditional tags don't take any parameters, but there are exceptions. Visit the WordPress Codex to learn more.

There's more...

Conditional tags, as we have seen, are very useful. Moreover, some conditional tags accept optional arguments that allow you to create more complex logic. Some conditional tags can accept up to four different types of parameters:

1. ID: The ID parameter can be used in `is_category`, `is_page()`, `is_tag()`, `is_single()`, `is_author()`, and `is_sticky()`.

   ```
   <?php if (is_category(5))
   {
   }?>
   ```

   As a result, the preceding code returns true if the category ID is `5`.

2. Name: The name parameter can be used in `is_category`, `is_page()`, `is_single()`, and `is_author()`.

   ```
   <?php if (is_category("Blogging Tips"))
   {
   }?>
   ```

   As a result, the preceding code returns true if the category name is `"Blogging Tips"`.

3. Slug: The slug parameter can be used in `is_category`, `is_page()`, `is_tag()`, `is_single()`, and `is_author()`.

   ```
   <?php if (is_category("blogging-tips"))
   {
   }?>
   ```

   As a result, the preceding code returns true if the category slug is `"blogging-tips"`.

4. Array: The array parameter can be used in `is_category`, `is_page()`, `is_tag()`, `has_tag()`, `is_single()`, and `is_author()`.

   ```
   <?php if(is_category(array(5,'blogging-tips','Blogging Tips')))
   {
   } ?>
   ```

   As a result, the preceding code returns true if the category of posts being displayed either has the `ID` of `5`, slug as `"blogging-tips"`, or the name `"Blogging Tips"`.

WordPress features a large number of conditional tags that provide for a wide variety of conditions. All work as described above. Here is a list of some of the most useful conditional tags:

- `is_home()`: Returns `true` if the current page is the blog homepage
- `is_front_page()` : Returns `true` if the current page is the blog front page
- `is_single()` : Returns `true` if the current page is a single post template
- `is_page()`: Returns `true` if the current page is a page template
- `is_page_template("about.php")`: Returns `true` if a page template is currently being used
- `is_category()`: Returns `true` if the current page is a category template
- `in_category('4')`: Returns `true` if the current posts belongs to the specified category
- `is_tag()`:Returns `true` if the current page is a tag template
- `has_tag("wordpress")`: Returns `true` if the post has the tag specified in parameter
- `is_author()`:Returns `true` if the current page is an author archive
- `is_date()`:Returns `true` if the post or page is a date-based archive
- `is_year()`:Returns `true` if it's a yearly archive
- `is_month()`:Returns `true` if it's a monthly archive
- `is_day()`:Returns `true` if it's a daily archive
- `is_time()`:Returns `true` if an hourly, minutely, or secondly archive is being displayed
- `is_archive()`:Returns `true` true if the current page display any type of archives (time, author, tag, and so on)
- `is_search()`:Returns `true` if the current page displays search results
- `is_paged()`:Returns `true` if the current page is paged
- `is_404()`:Returns `true` if the current page is a 404
- `is_sticky()`:Returns `true` if the **Stick this post to the front page** check box has been checked for the current post
- `has_tag("WordPress")`: Returns `true` if the current posted has been tagged with **WordPress**
- `is_admin()`:Returns `true` if the dashboard or an admin page is currently displayed
- `comments_open()`:Returns `true` if commenting is allowed on the post
- `pings_open()`:Returns `true` if pinging is allowed on the post
- `is_preview()`:Returns `true` if the post or page is displayed in preview mode

[A complete and current list of WordPress conditional tags is maintained at `http://codex.wordpress.org/Conditional_Tags`]

Using multiple page templates

In WordPress themes, templates are used to control the layout available to your pages. Even though most of WordPress themes use a single page template, WordPress allows you to create as many different page templates as you want. Running multiple templates opens up a significant degree of flexibility in design and page layouts. You could, for example, create a page template specifically suited for forms, or others tailored to the page content and functionality.

In this recipe, you'll learn to create and use page templates.

Getting ready

To achieve this recipe, all you need is your favorite text editor and access to the WordPress files on your server.

How to do it...

We're going to start out with a bare minimum page template; once you've set up the basics, you can style it anyway you like. To create a basic page template, follow these steps:

1. Access your WordPress installation on your server.
2. Navigate to your active theme directory and find the file `page.php`.
3. Make a copy of the file and rename it `mytemplate.php`.
4. Open the `mytemplate.php` file in your favorite text editor and find the following code, located at the very top of the file:

```
<?php
/**
 * The template for displaying all pages.
*
* This is the template that displays all pages by default.
* Please note that this is the WordPress construct of pages
* and that other 'pages' on your WordPress site will use a
* different template.
*
* @package WordPress
* @subpackage Twenty_Eleven
* @since Twenty Eleven 1.0
*/
```

5. Replace the code above with this code:

    ```
    <?php
    /**
    * My first custom template
     *
     * Template Name: My Template
     *

    */
    ```

6. Save the file.

That's all there is to it; your new page template is ready to use. To apply it to a page on your site, follow these steps:

1. Login to your WordPress **Dashboard.**
2. Click on the **Pages** menu.
3. Open any of the pages for editing by clicking the **Edit** link, seen when you move your over the page listing.
4. Look to the right column where you can see a combo box labeled **Template**.
5. Click to open the **Template** drop-down menu and you will see listed there your new template, **My Template**.
6. Select **My Template**.
7. Click on the **Update** button to apply the change.

If you visit your page, you will now see your new template applied.

How it works...

The WordPress page template allows you to define special template layouts that can be assigned to pages. As the administrator creates (or edits) a page, a template can be selected from the **Page Template** combo box in the sidebar.

> Your custom page template files are also available for editing through the WordPress' built-in editor, under the **Appearance** menu.

There's more...

In the steps above, we created the most basic of page templates. However, now that you know how to make a page template, you can do with it whatever you want. Add your own styling, hard code text, or images, even add your own functional elements.

There's no upper limit on how many page templates you can create, so you can feel free to make any number of purpose-built templates that can be used by your content creators.

> For some ideas of what you can do with this powerful WordPress feature, visit this page `http://codex.wordpress.org/Pages#Page_Templates`

Adding functionality with template tags

Template tags in WordPress work like snippets that provide you with discrete bits of functionality, but without the hassle of dealing with long segments of code. The tags are used to display information dynamically or to customize the output of the page.

There are a number of template tags available to you. You can see a complete list by visiting `http://codex.wordpress.org/Template_Tags`. Note that some tags include parameters that let you further extend the functionality.

Implementing a template tag is simply a matter of wrapping the tag in a php statement. Where parameters are available, you can also declare those, if you wish to use them. For example, let's say that in a particular page template you wish to automatically insert the date the page was posted. You could do this very easily by employing a template tag: `the_date()`. Your code might look like this: `<p>Date created: <?php the_date(); ?></p>`

> In the example above, I wrapped the template tag in a `<p>` tag, but you could use the styling of your choice or even omit it completely.

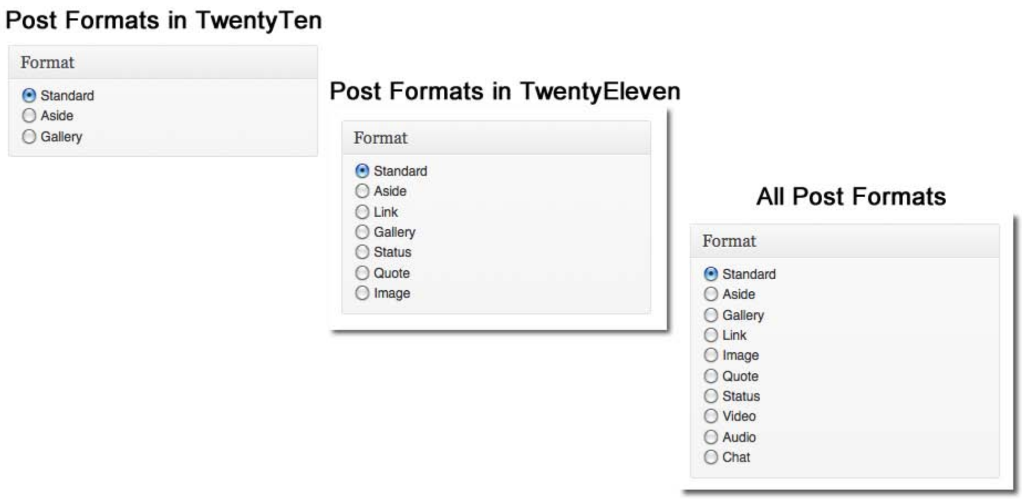
There are other options as well, since `the_date()` also supports the following parameters: `$format`, `$before`, `$after`, `$echo`. Using the parameters, you could, for example, specify that the date will be in the format Year-Month-Day and also be wrapped with an `<H3>` tag: `<?php the_date('Y-m-d', '<h3>', '</h3'); ?>`

> You can download a handy template tags cheatsheet by visiting `http://docs.ekinertac.com/Wordpress-Cheat-Sheet.pdf`

Using post formats

Post formats are a relatively new feature in WordPress, having first appeared in WordPress 3.1. The formats are a welcome development. In the past, many developers used categories as a way to group post content by nature or function and had to style based on category. With post formats, you have available a wide range of options that cover the most common categories of use, including audio, video, images, and galleries.

Support for post formats depends on the theme. As you can see in the next screenshot, there are even differences between **TwentyTen** and **TwentyEleven**. While **TwentyEleven** comes ready to use with a large number of post formats, in reality the theme uses only seven of the ten formats WordPress provides.


Post Formats in TwentyTen
Format
● Standard
○ Aside
○ Gallery
Post Formats in TwentyEleven
Format
● Standard
○ Aside
○ Link
○ Gallery
○ Status
○ Quote
○ Image
All Post Formats
Format
● Standard
○ Aside
○ Gallery
○ Link
○ Image
○ Quote
○ Status
○ Video
○ Audio
○ Chat

In this recipe we're going to look at how you can enable some or all of the additional post formats in your theme.

Getting ready

Post formats are a default feature of WordPress, however support for them varies from theme to theme. In this recipe we will look at the default Twenty Ten theme. We will modify the functions`.php` file to provide more post formats for the theme. To complete this recipe you will need a code editor and access to your WordPress installation on your server.

How to do it...

1. Access your WordPress installation on your server.
2. Open the `functions.php` file of your active theme. In this example, we are using the `functions.php` file from the default TwentyTen theme.
3. Find the following lines in the file:

```
function twentyten_setup() {



   // This theme styles the visual editor with editor-style.css to
match the theme style.

   add_editor_style();

   // Post Format support. You can also use the legacy "gallery"
or "asides" (note the plural) categories.

   add_theme_support( 'post-formats', array( 'aside', 'gallery' )
);
```

4. We want to add additional items to the array of post formats, so we need to modify the last line of that code as follows:

```
add_theme_support( 'post-formats', array( 'aside',
  'gallery','link','image','quote','status','video','audio','chat'
) );
```

5. Save the file.

If you now access your WordPress dashboard and create a new post, or edit an existing post, you will see a full list of options under the **Format** sidebar, as seen in the preceding image.

How it works...

WordPress comes with a default set of post formats. While you cannot add your own, you can style the ones that are included in the system. The default post formats are:

- `Aside`: For holding a short piece of content, typically without a title
- `Audio`: To hold a single audio file
- `Chat`: Formatted to hold a chat transcript
- `Gallery`: Intended for use as a gallery of images
- `Image`: A single image file
- `Link`: A single link (URL) to an external site
- `Quote`: A quotation, often done without a title
- `Status`: A short status message, like you would see on Twitter or another micro-blogging service
- `Video`: Holds a single video file

Additionally there is the `standard` post format, which is simply the default format used by the system if no other format is specified.

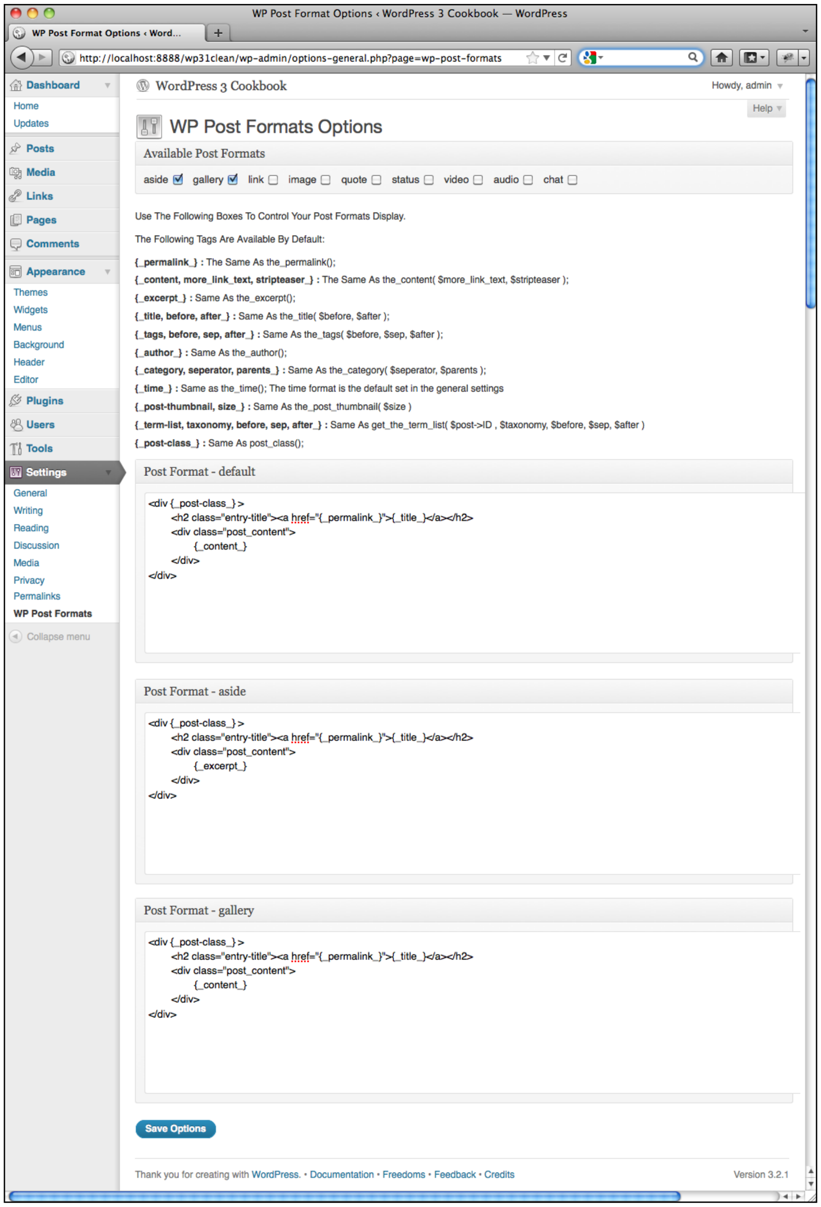
The post format options are set in the theme by the init function. In the recipe, we simply accessed the `functions.php` file, found the relative line, and then added to the array of post format types. Now, when the theme is activated, all post formats will be available through the WordPress post creation and editing screens.

There's more...

The recipe above shows you how to enable all the default post formats. The next question for most people will be "how do I style the post formats"?

An easy option for styling post formats

Styling post formats can be rather clunky, involving diving into the WordPress loop. There is, however, another way. The WP Post Formats plugin makes enabling and customizing post formats easy. Once installed, the plugin exposes the entire range of post format options and also includes editing windows that let you manage template for each of the formats, as you can see in the following screenshot :


WP Post Format Options ‹ WordPress 3 Cookbook — WordPress
WP Post Format Options ‹ Word...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=wp-post-formats
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Background
Header
Editor
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
WP Post Formats
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help

WP Post Formats Options

Available Post Formats

aside ☑ gallery ☑ link ☐ image ☐ quote ☐ status ☐ video ☐ audio ☐ chat ☐

Use The Following Boxes To Control Your Post Formats Display.

The Following Tags Are Available By Default:

{_permalink_} : The Same As the_permalink();
{_content, more_link_text, stripteaser_} : The Same As the_content($more_link_text, $stripteaser);
{_excerpt_} : Same As the_excerpt();
{_title, before, after_} : Same As the_title($before, $after);
{_tags, before, sep, after_} : Same As the_tags($before, $sep, $after);
{_author_} : Same As the_author();
{_category, seperator, parents_} : Same As the_category($seperator, $parents);
{_time_} : Same as the_time(); The time format is the default set in the general settings
{_post-thumbnail, size_} : Same As the_post_thumbnail($size)
{_term-list, taxonomy, before, sep, after_} : Same As get_the_term_list($post->ID , $taxonomy, $before, $sep, $after)
{_post-class_} : Same As post_class();

Post Format - default

```
<div {_post-class_} >
    <h2 class="entry-title"><a href="{_permalink_}">{_title_}</a></h2>
    <div class="post_content">
        {_content_}
    </div>
</div>
```

Post Format - aside

```
<div {_post-class_} >
    <h2 class="entry-title"><a href="{_permalink_}">{_title_}</a></h2>
    <div class="post_content">
        {_excerpt_}
    </div>
</div>
```

Post Format - gallery

```
<div {_post-class_} >
    <h2 class="entry-title"><a href="{_permalink_}">{_title_}</a></h2>
    <div class="post_content">
        {_content_}
    </div>
</div>
```

Save Options
Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits
Version 3.2.1

Creating a custom 404 error page

404, or 'page not found' errors, are part of life on the web. They are impossible to avoid. What you can do, however, is control what kind of an impact they have on your site's users. Graceful management of 404 errors is one of the hallmarks of a professional website. When handled properly, the number of visitors you will lose due to 404 errors will drop off substantially.

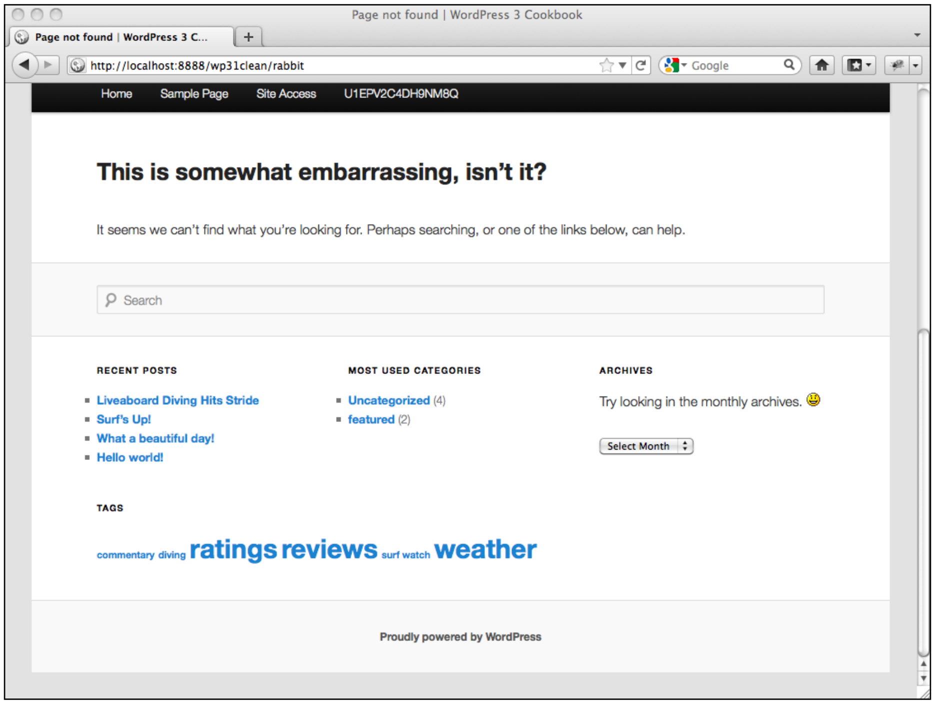
It is important that you site has a proper 404 page. The key aspects here are making sure the 404 error page shown is consistent with your brand and identity and providing your site visitors with a way to reach the rest of your site. In this recipe we discuss how to modify the default 404 page, if your site's theme already has one, and if not, how to create one.

Getting ready

For this recipe, you'll need a code editor and access to the files of your WordPress installation.

How to do it...

Many WordPress themes, including the default Twenty Eleven, come with a special template to handle 404 errors. The default Twenty Eleven 404 page template is displayed in the following screenshot:


Page not found | WordPress 3 Cookbook
Page not found | WordPress 3 C...
http://localhost:8888/wp31clean/rabbit
Google
Home
Sample Page
Site Access
U1EPV2C4DH9NM8Q
Search
RECENT POSTS
MOST USED CATEGORIES
ARCHIVES
Select Month
TAGS

This is somewhat embarrassing, isn't it?

It seems we can't find what you're looking for. Perhaps searching, or one of the links below, can help.

RECENT POSTS
- Liveaboard Diving Hits Stride
- Surf's Up!
- What a beautiful day!
- Hello world!

MOST USED CATEGORIES
- Uncategorized (4)
- featured (2)

ARCHIVES

Try looking in the monthly archives. 😃

TAGS

commentary diving ratings reviews surf watch weather

Proudly powered by WordPress

The 404 error template is named `404.php` and resides in the active theme directory. If you wish to edit this file, follow these steps:

1. Access your WordPress installation.
2. Go to the directory that contains your active theme.
3. Open the `404.php` file with your favorite editor.
4. Make any changes you wish to the file.
5. Save the file back to your server, overwriting the original.

If, for some reason, your theme doesn't include a `404.php`, you should create one. To create simple 404 page, follow these steps:

1. Create a new file with your favorite editor.
2. Name it `404.php`
3. Paste the following code in the file:

   ```
   <?php get_header(); ?>
       <div id="content" class="narrowcolumn">
       <h2 class="center">Error 404 - Not Found</h2>
       </div>
   <?php get_sidebar(); ?>
   <?php get_footer(); ?>
   ```

4. Save the file to your active theme's directory, and you're done!

How it works...

WordPress features an interesting template hierarchy mechanism. When a 404 error appears, WordPress automatically looks for a file named `404.php`. If such a file exists, it is displayed.

Using a static page as a homepage

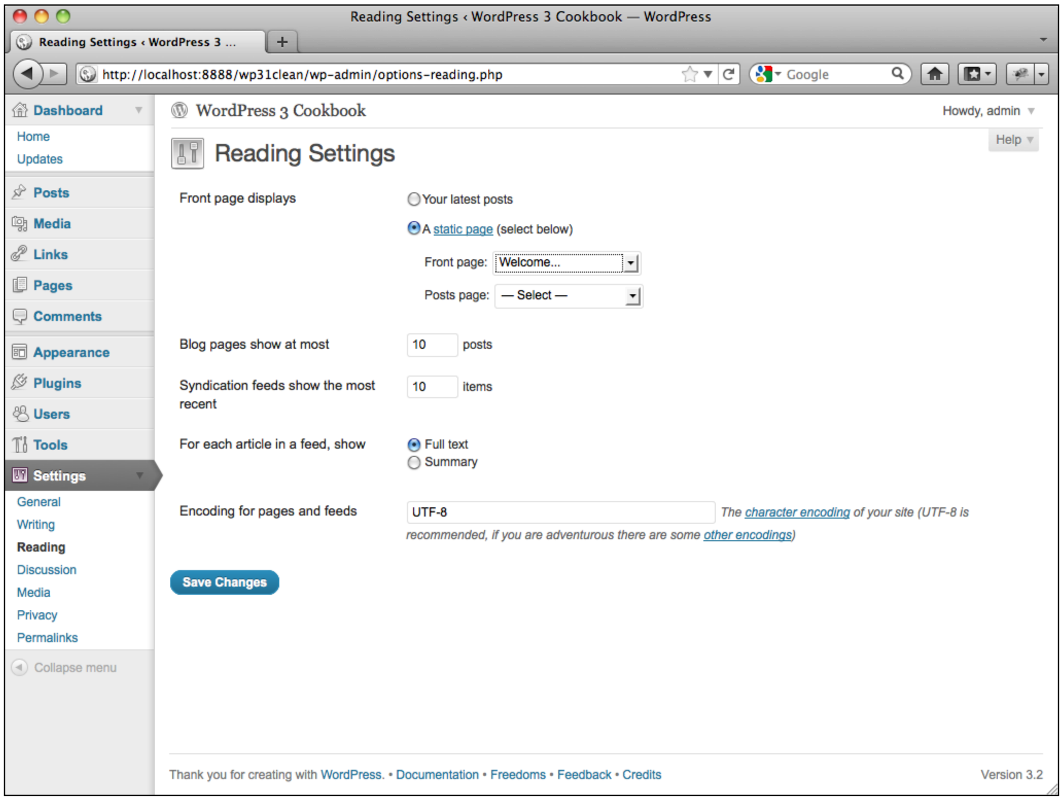
Quite a few WordPress sites use the list of most recent posts as the site homepage. This is the default option and a common solution for blog sites. However, if you prefer to use a static page as a homepage, you can do so easily. In this recipe, we will learn how to set up a static page as a homepage for your site.

Getting ready

Nothing special is needed here. The possibility of using a static page as a homepage is built-in in the WordPress.

How to do it...

1. Log in to your WordPress **Dashboard.**
2. Click on the **Pages** menu.
3. Click on the **Add New** option.
4. Create a new page for use as your site's homepage. Name it **Welcome...**
5. Publish your new **Welcome...** page.
6. Click on the **Settings** menu.
7. Click on the option **Reading**.
8. You'll see a title saying **Front page displays**. Select the radio button marked **A static page**.
9. Choose your new front page from the drop-down list marked **Front page:**. The following screenshot shows the settings:


Reading Settings ‹ WordPress 3 Cookbook — WordPress
Reading Settings ‹ WordPress 3 ...
http://localhost:8888/wp31clean/wp-admin/options-reading.php
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
Save Changes

Reading Settings

Front page displays ○ Your latest posts ● A static page (select below)

Front page: Welcome...

Posts page: — Select —

Blog pages show at most 10 posts

Syndication feeds show the most recent 10 items

For each article in a feed, show ● Full text ○ Summary

Encoding for pages and feeds UTF-8 The character encoding of your site (UTF-8 is recommended, if you are adventurous there are some other encodings)

Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits Version 3.2

10. Click the **Save Changes** button.

How it works...

WordPress allows you to choose your home page content. You can use either the list of posts, or a static page of your choice. This second option gives you quite a few possibilities, in particular if you'd like to create a non-blog site using WordPress.

Adding custom styles to your theme

Now that you know how to create a child theme and are able to use post formats and custom page templates, you will need to be able to insert new styling into your themes. The basic principles are straight out of CSS 101. In this recipe we look at the simple steps needs to add a custom style to your theme.

Getting ready

We're going to do everything in this recipe using only the tools inside your WordPress dashboard. For the sake of simplicity, we're using the default TwentyEleven theme. The principles demonstrated will work for anything, though the code may vary.

How to do it...

There are two parts to this recipe: First, you need to wrap the area you want to affect with the styling. Second, you need to define the CSS selector in the theme's stylesheet. For this example, let's modify the header of the page to add new div we can use to control the area around the top menu:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.
3. Click on the **Editor** option.
4. Find the **Template** named **Header** and click on it.
5. Find the tag `<hgroup>`
6. Let's add some styling to this area. Modify the tag to look like this: `<hgroup id="headhgroup">`
7. Click the **Update File** button.

That completes the first step. The new style has been added and whatever we define for `#headhgroup` will now impact the area wrapped by the `<hgroup>` tag. Let's now add a definition for the selector:

1. Click on **Stylesheet**.

2. Scroll to the bottom of the file and let's add a new selector:

   ```
   #headhgroup {
      background: #666666;
   }
   ```

3. Click on the **Update File** button.

If you visit the front end of your website now and refresh the page in your browser, you should see a light gray colored background in the head of your pages.

How it works...

We're simply using the power of HTML and CSS. We've defined a selector in the theme's stylesheet, then applied it to a section of the code in the `header.php` file.

Making your site mobile device friendly

With the ubiquity of handheld and portable devices and with ready access to WiFi and other connectivity, having a mobile-friendly website is now almost a necessity. While you can go to the effort of building a dedicated mobile site or creating a mobile theme, a new WordPress plugin makes it easy to have a simple and effective mobile-friendly website.

In this recipe, we look at using the WPtouch plugin.

> In this example we use the free version of the plugin. There is also a commercial version. Though the free version is fully functional, you will want to look at all the options.

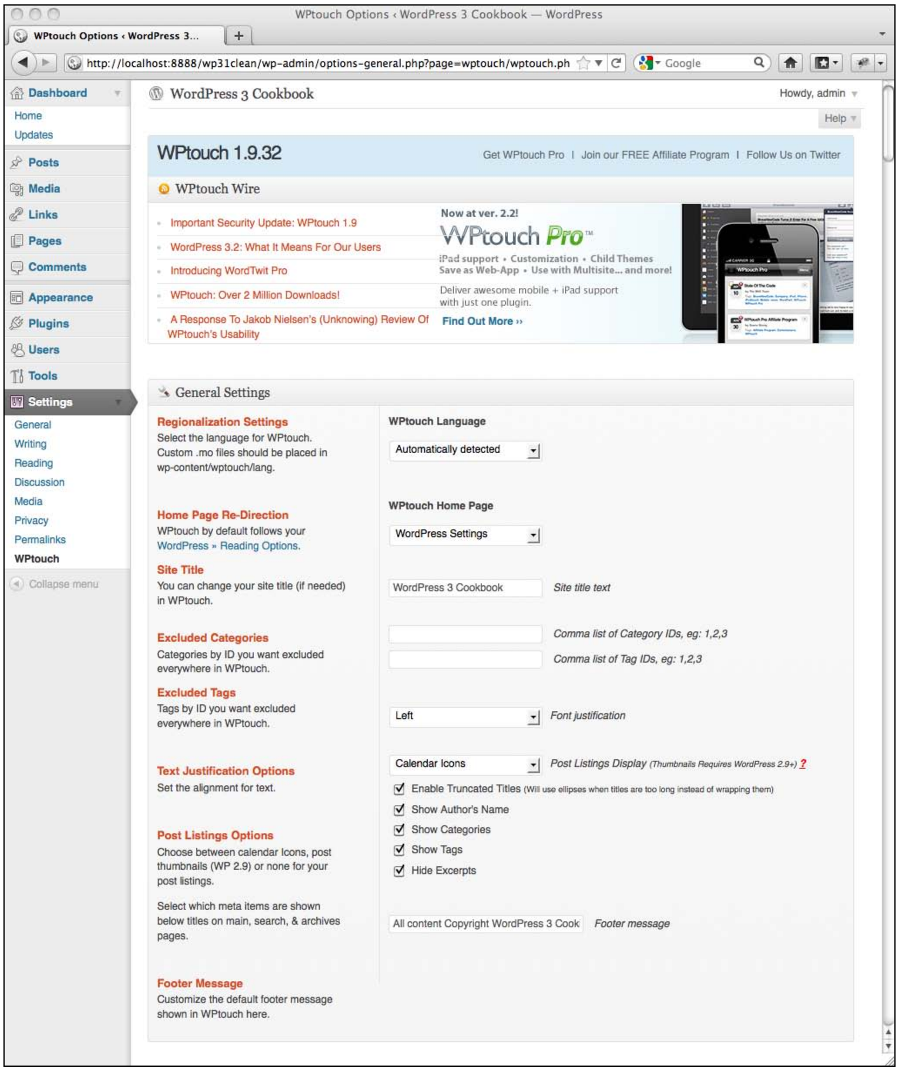
Getting ready

To execute this recipe, you will also need to install the WPtouch plugin. You will need to install this plugin before you can get started. Search for **WPtouch** inside the **Add New Plugins** screen of your WordPress site. After you find it, click to install, and then activate it.

> You can learn more about this plugin by visiting the plugin's page on Wordpress.org at `http://wordpress.org/extend/plugins/wptouch/`.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the option **WPtouch**.
4. In the **General Settings** portion of the page, seen in the screenshot below, you can set the basic configuration options you need for the plugin to display your site. The numerous advanced options seen further down the **WPtouch** configuration page allow you to tailor things more closely to your needs.


WPtouch Options ‹ WordPress 3 Cookbook — WordPress
WPtouch Options ‹ WordPress 3...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=wptouch/wptouch.ph
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
WPtouch
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
WPtouch 1.9.32
Get WPtouch Pro | Join our FREE Affiliate Program | Follow Us on Twitter
WPtouch Wire
Important Security Update: WPtouch 1.9
WordPress 3.2: What It Means For Our Users
Introducing WordTwit Pro
WPtouch: Over 2 Million Downloads!
A Response To Jakob Nielsen's (Unknowing) Review Of WPtouch's Usability
Now at ver. 2.2!
WPtouch Pro™
iPad support • Customization • Child Themes
Save as Web-App • Use with Multisite... and more!
Deliver awesome mobile + iPad support with just one plugin.
Find Out More ››
General Settings

Regionalization Settings
Select the language for WPtouch. Custom .mo files should be placed in wp-content/wptouch/lang.

WPtouch Language: Automatically detected

Home Page Re-Direction
WPtouch by default follows your WordPress » Reading Options.

WPtouch Home Page: WordPress Settings

Site Title
You can change your site title (if needed) in WPtouch.

WordPress 3 Cookbook — *Site title text*

Excluded Categories
Categories by ID you want excluded everywhere in WPtouch.

Comma list of Category IDs, eg: 1,2,3

Excluded Tags
Tags by ID you want excluded everywhere in WPtouch.

Comma list of Tag IDs, eg: 1,2,3

Text Justification Options
Set the alignment for text.

Left — *Font justification*

Post Listings Options
Choose between calendar Icons, post thumbnails (WP 2.9) or none for your post listings.

Select which meta items are shown below titles on main, search, & archives pages.

Calendar Icons — *Post Listings Display (Thumbnails Requires WordPress 2.9+)* ?

☑ Enable Truncated Titles (Will use ellipses when titles are too long instead of wrapping them)
☑ Show Author's Name
☑ Show Categories
☑ Show Tags
☑ Hide Excerpts

Footer Message
Customize the default footer message shown in WPtouch here.

All content Copyright WordPress 3 Cook — *Footer message*

5. When you've finished, click **Save Options**.

Now, when a visitor accesses your site from a mobile device, WPtouch will automatically display the contents in a mobile-friendly format.

> The WPtouch plugin has been tested with iPhone, iPod Touch, Google Android, Blackberry Storm, Blackberry Torch, Palm Pre, and Samsung touch.

How it works...

Though WPtouch seems a great deal like a theme, the detection of the mobile device and the formatting of the contents on the fly is the byproduct of a plugin. When a mobile device visits the site, the plugin automatically detects it and feeds that device a different set of theme files. As a result, the installation of this plugin will have no impact on the appearance of the site in a browser on a standard resolution monitor.

3

Working with Plugins and Widgets

In this chapter, we will cover:

- Installing plugins
- Installing widgets
- Adding new widget areas to your theme
- Creating your own widgets
- Modifying core widgets
- Displaying tabs on your sidebar
- Using conditional tags to control widget display
- Displaying widgets inside of posts and pages

Introduction

Plugins are the key to extending the functionality of your WordPress site. Virtually all WordPress installations run one or more plugins; many run quite a few. Plugins vary in complexity, from adding very simple functionality to very complex plugins that require extensive configuration.

Widgets work hand-in-hand with plugins by allowing you to position discrete bits of simple functionality on your pages. Widgets are often a by-product of plugins; when you install a new plugin, you will often get new widgets that can be used to expose all or part of the plugin's functionality.

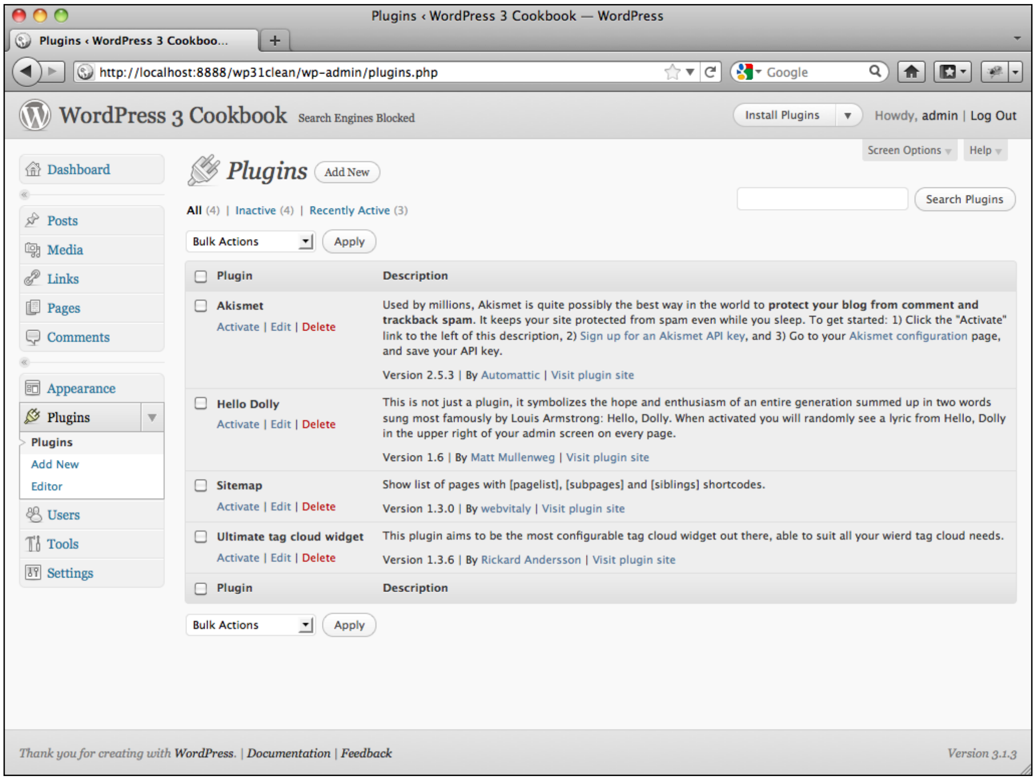
In this chapter, we look at how to extend your site through the strategic use of plugins and widgets.

Installing plugins

The first step to adding new functionality to your site is to learn how to install new plugins for your WordPress site.

In this recipe we look at how to install plugins. Installation can be either automatic or manual; both are discussed in this recipe.

The key interface for plugin management, including installation, is the Plugins Manager, shown in the next screenshot. The Plugins Manager exists in all WordPress sites and can be accessed by clicking on the **Plugins** menu.


Plugins ‹ WordPress 3 Cookbook — WordPress
Plugins ‹ WordPress 3 Cookboo...
http://localhost:8888/wp31clean/wp-admin/plugins.php
Google
WordPress 3 Cookbook
Search Engines Blocked
Install Plugins
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Settings
Plugins
Add New
Search Plugins
All (4) | Inactive (4) | Recently Active (3)
Bulk Actions
Apply
Bulk Actions
Apply
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

☐ Plugin	Description
☐ **Akismet** Activate | Edit | Delete	Used by millions, Akismet is quite possibly the best way in the world to **protect your blog from comment and trackback spam**. It keeps your site protected from spam even while you sleep. To get started: 1) Click the "Activate" link to the left of this description, 2) Sign up for an Akismet API key, and 3) Go to your Akismet configuration page, and save your API key. Version 2.5.3 | By Automattic | Visit plugin site
☐ **Hello Dolly** Activate | Edit | Delete	This is not just a plugin, it symbolizes the hope and enthusiasm of an entire generation summed up in two words sung most famously by Louis Armstrong: Hello, Dolly. When activated you will randomly see a lyric from Hello, Dolly in the upper right of your admin screen on every page. Version 1.6 | By Matt Mullenweg | Visit plugin site
☐ **Sitemap** Activate | Edit | Delete	Show list of pages with [pagelist], [subpages] and [siblings] shortcodes. Version 1.3.0 | By webvitaly | Visit plugin site
☐ **Ultimate tag cloud widget** Activate | Edit | Delete	This plugin aims to be the most configurable tag cloud widget out there, able to suit all your wierd tag cloud needs. Version 1.3.6 | By Rickard Andersson | Visit plugin site
☐ Plugin	Description

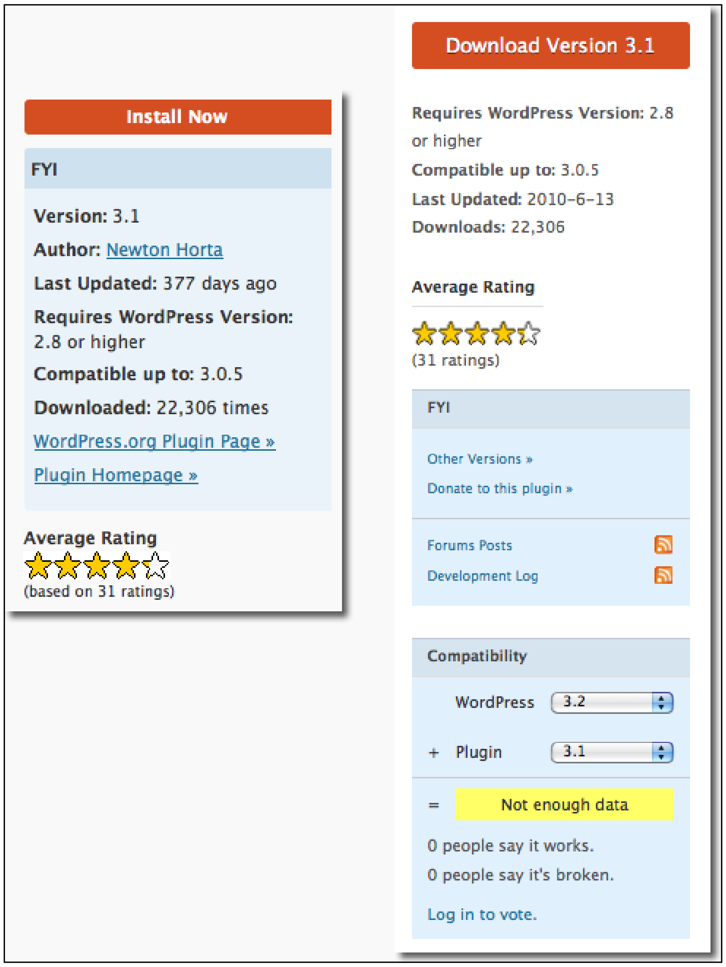
Getting ready

As noted above, installing a WordPress plugin can be done one or two ways: either manually or automatically. If you use the automatic installer, seen at the top of the Plugins Manager in the preceding screenshot, then you need no additional tools to complete the recipe. If on the other hand, you wish to install a plugin manually, you will need the ability to add files to your WordPress installation on your server.

> You can find new plugins by either searching from inside the Plugins Installer in the WordPress dashboard (as explained below), or by visiting `www.Wordpress.org` at `http://wordpress.org/extend/plugins/`

Before you install any plugin, you must first check the compatibility of the plugin. If the plugin is not cleared to work with your version of WordPress, you should be very reluctant to install it. If the plugin is not approved for your version of WordPress, it may not function properly or it may introduce security risks.

To identify the compatibility of a plugin, you can either view the plugin's page on the WordPress.org site, or you can click and view details about the plugin from inside the plugin installer in the Plugins Manager. The following screenshot shows how the version data appears in both situations:



Install Now
FYI

Version: 3.1

Author: Newton Horta

Last Updated: 377 days ago

Requires WordPress Version: 2.8 or higher

Compatible up to: 3.0.5

Downloaded: 22,306 times

WordPress.org Plugin Page »
Plugin Homepage »
Average Rating

★★★★☆

(based on 31 ratings)

Download Version 3.1

Requires WordPress Version: 2.8 or higher

Compatible up to: 3.0.5

Last Updated: 2010-6-13

Downloads: 22,306

Average Rating

★★★★☆

(31 ratings)

FYI
Other Versions »
Donate to this plugin »
Forums Posts
Development Log
Compatibility

WordPress 3.2

+ Plugin 3.1

= Not enough data

0 people say it works.

0 people say it's broken.

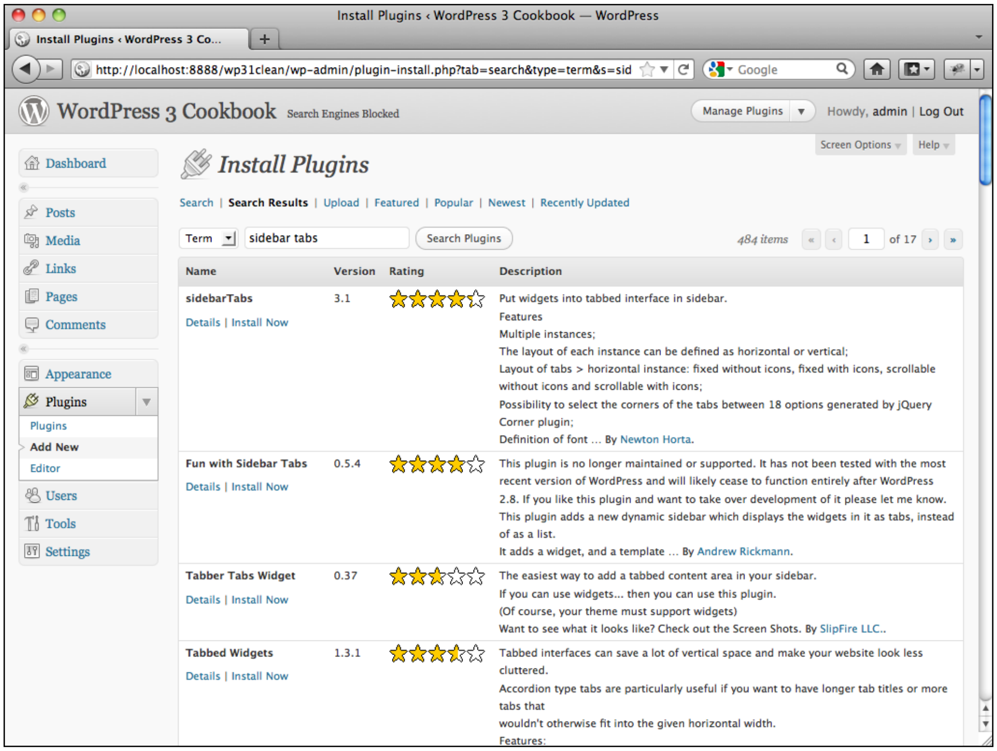
Log in to vote.

On the left you can see the version information presented in the Plugins Installer inside the WordPress dashboard. On the right is the information as it appears on `www.WordPress.org`.

How to do it...

To install a plugin automatically, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Plugins** menu.
3. Click on the option **Add New**.
4. On the screen that appears, enter a search term in the **Search** field.
5. Click on the **Search Plugins** button.
6. On the search results screen, shown below, review the options. You can click on the **Details** link to see a pop-up description of the plugin, including the compatibility information.
7. Once you find the plugin you want, click on the **Install Now**.
8. Once the system completes installation, you will be prompted to **Activate** the plugin. You can either do so now, or at a later point from the Plugins Manager.


Install Plugins ‹ WordPress 3 Cookbook — WordPress
Install Plugins ‹ WordPress 3 Co...
http://localhost:8888/wp31clean/wp-admin/plugin-install.php?tab=search&type=term&s=sid
Google
WordPress 3 Cookbook
Search Engines Blocked
Manage Plugins
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Settings
Install Plugins
Search | Search Results | Upload | Featured | Popular | Newest | Recently Updated
Term
sidebar tabs
Search Plugins
484 items
1
of 17

Name	Version	Rating	Description
sidebarTabs Details | Install Now	3.1	★★★★☆	Put widgets into tabbed interface in sidebar. Features Multiple instances; The layout of each instance can be defined as horizontal or vertical; Layout of tabs > horizontal instance: fixed without icons, fixed with icons, scrollable without icons and scrollable with icons; Possibility to select the corners of the tabs between 18 options generated by jQuery Corner plugin; Definition of font ... By Newton Horta.
Fun with Sidebar Tabs Details | Install Now	0.5.4	★★★★☆	This plugin is no longer maintained or supported. It has not been tested with the most recent version of WordPress and will likely cease to function entirely after WordPress 2.8. If you like this plugin and want to take over development of it please let me know. This plugin adds a new dynamic sidebar which displays the widgets in it as tabs, instead of as a list. It adds a widget, and a template ... By Andrew Rickmann.
Tabber Tabs Widget Details | Install Now	0.37	★★★☆☆	The easiest way to add a tabbed content area in your sidebar. If you can use widgets... then you can use this plugin. (Of course, your theme must support widgets) Want to see what it looks like? Check out the Screen Shots. By SlipFire LLC..
Tabbed Widgets Details | Install Now	1.3.1	★★★½☆	Tabbed interfaces can save a lot of vertical space and make your website look less cluttered. Accordion type tabs are particularly useful if you want to have longer tab titles or more tabs that wouldn't otherwise fit into the given horizontal width. Features:

To install a plugin manually, follow these steps:

1. Obtain a copy of the archive file containing your plugin; typically this is done from the `www.WordPress.org` Plugins page.
2. Extract the plugin locally on your computer.
3. Access the filesystem of your WordPress installation.
4. Navigate to the directory `/wp-content/plugins`.
5. Copy the directory containing your plugin files to the directory on your server.
6. Log in to your WordPress **Dashboard**.
7. Click on the **Plugins** menu.
8. Scan the list of plugins to find the name of the plugin you have just installed.
9. Click on **Activate** to use the plugin.

At this point, your plugin should be ready to go. At any time, you can visit the Plugins Manager to disable or delete your plugin.

How it works...

When a plugin is installed, its name is registered in the WordPress database, which means that the plugin is active and WordPress will let it execute the tasks it was developed for.

Plugins use hooks (we will discuss this later) and the Plugin API, which provide useful PHP functions to modify WordPress default functions.

There's more...

If you decide to stop using a plugin you should de-activate it and delete it from your system. Leaving unused plugins on your site not only increases your maintenance burden, but also exposes your site to security risk. Moreover, in some situations, you may install a plugin only to find that it does not work properly, or even worse, causes a problem for your site. If that happens, you will need to remove the plugin.

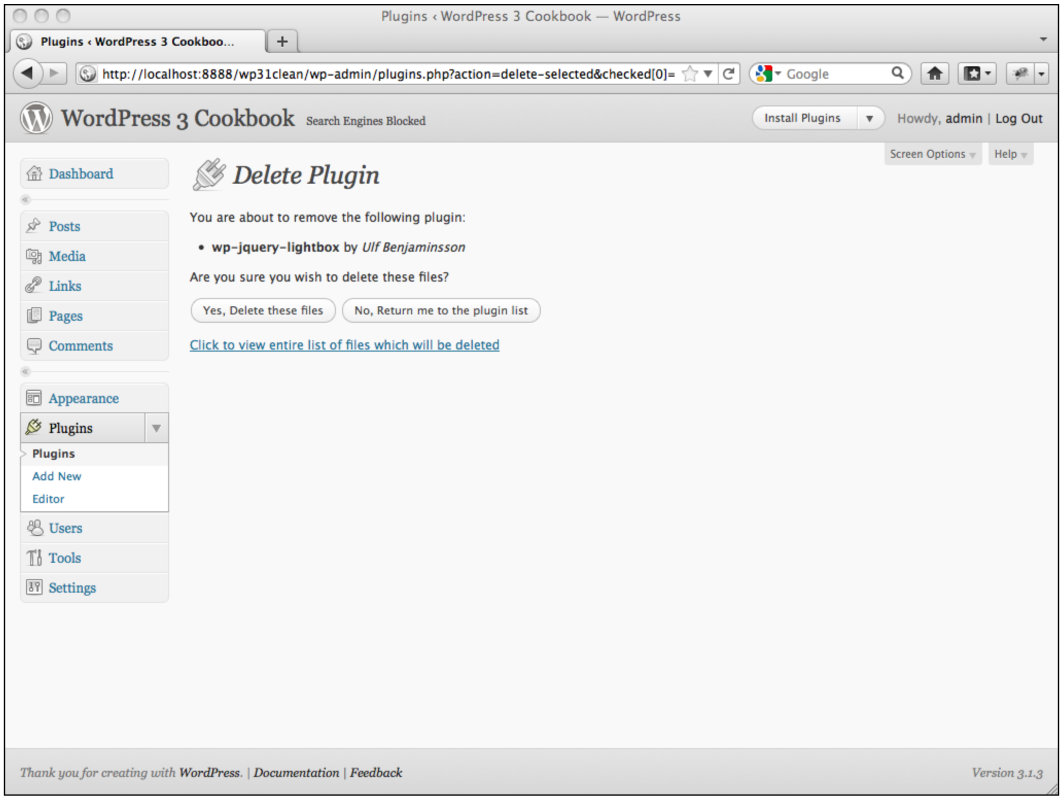
In WordPress, deleting a plugin is very easy, and you can always re-install it again if you decide to use it in the future.

Deleting installed plugins automatically

The following steps can be used to delete a plugin automatically:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Plugins** menu.
3. Scan the list of plugins to find the name of the plugin you want to remove.

4. Make sure the plugin is de-activated. If it is still active, then click on the **Deactivate** link.
5. Once deactivated, you can remove the plugin completely by clicking on the **Delete** link.
6. After you click the link, you will be prompted to confirm the decision, as seen in the following screenshot:


Plugins ‹ WordPress 3 Cookbook — WordPress
Plugins ‹ WordPress 3 Cookboo...
http://localhost:8888/wp31clean/wp-admin/plugins.php?action=delete-selected&checked[0]=
Google
WordPress 3 Cookbook
Search Engines Blocked
Install Plugins
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Settings
Yes, Delete these files
No, Return me to the plugin list

Delete Plugin

You are about to remove the following plugin:

- **wp-jquery-lightbox** by *Ulf Benjaminsson*

Are you sure you wish to delete these files?

Click to view entire list of files which will be deleted

Thank you for creating with WordPress. | Documentation | Feedback

Version 3.1.3

7. To confirm the delete, click on the **Yes, Delete these files** button. To cancel and return to the Plugins Manager, click on the **No, Return me to the plugin list** button.

Deleting installed plugins manually

The following steps can be used to delete a plugin from your system manually:

1. Access the filesystem of your WordPress installation.
2. Navigate to the directory `/wp-content/plugins`.
3. Find the directory containing the file of the plugin you need to disable.
4. Rename the directory, which will force the system to deactivate the plugin.
5. Check your site; if it is working properly, you may now delete the directory containing the plugin's files.
6. Log in to your WordPress **Dashboard**.
7. Click on the **Plugins** option on the menu.
8. Scan the list of plugins to see if the plugin has been removed.

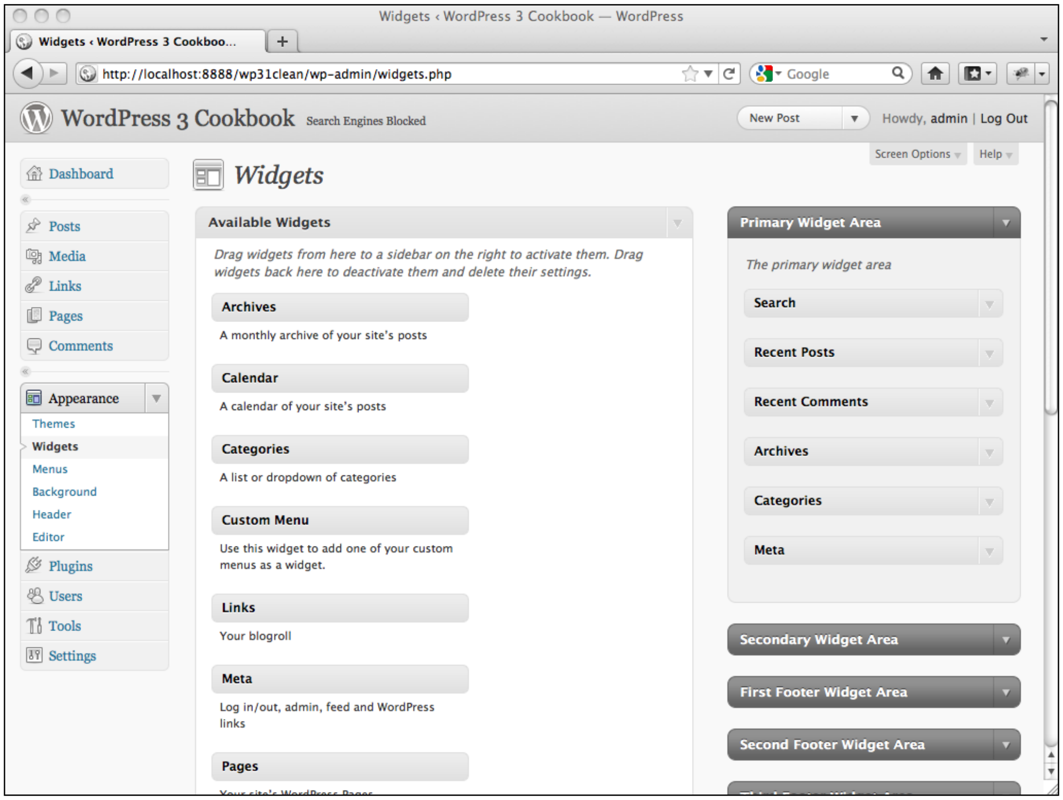
> Some plugins modify the WordPress database. If you wish to completely remove plugins that modify the database, you will need to also access your site's database, identify the information inserted by the plugin, and then remove it from the database. Check the plugin's documentation to determine whether this is necessary.

Installing widgets

Now that we have studied how to install WordPress plugins and what plugins can do for you, let's have a look at widgets. A widget does the same thing as a plugin, that is, it adds more functionality to your site. Widget installation, however, works differently than plugin installation. Typically, widgets are by-products of plugins. Widgets are not created independently and installed independently.

To use widgets on your site, you do have to take some steps. Assuming that your theme is widget-ready, you can visit the Widgets Manager screen, shown in the next screenshot, and simply drag a widget to a widget-ready zone.

Most all themes are widget-ready, meaning that the theme has defined zones into which you can manually place widgets from within the Widgets Manager. If your theme does not have pre-defined zones into which you can add widgets, you will need to modify the theme's files to add this functionality. Modifying your theme to add new widget areas is discussed below in the section *Adding Widget Areas to your Theme*.


Widgets ‹ WordPress 3 Cookbook — WordPress
Widgets ‹ WordPress 3 Cookboo...
http://localhost:8888/wp31clean/wp-admin/widgets.php
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Background
Header
Editor
Plugins
Users
Tools
Settings
Widgets
Available Widgets
Archives
Calendar
Categories
Custom Menu
Links
Meta
Pages
Primary Widget Area
Search
Recent Posts
Recent Comments
Archives
Categories
Meta
Secondary Widget Area
First Footer Widget Area
Second Footer Widget Area

Drag widgets from here to a sidebar on the right to activate them. Drag widgets back here to deactivate them and delete their settings.

Archives: A monthly archive of your site's posts

Calendar: A calendar of your site's posts

Categories: A list or dropdown of categories

Custom Menu: Use this widget to add one of your custom menus as a widget.

Links: Your blogroll

Meta: Log in/out, admin, feed and WordPress links

The primary widget area

Getting ready

There are no special requirements to complete this recipe. Everything you need is found inside the WordPress dashboard.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.

3. Click on the **Widgets** option.
4. On the Widgets Manager page, find the widget you wish to use.
5. Click and drag the widget to any of the widget areas you see on the right hand column.
6. Click the arrow on the title bar of the widget to open the widget and view the configuration options.
7. Edit its parameters as desired.
8. Click on the **Save** button on the widget and you're done!

See also

- *Adding widget areas to your theme*
- *Using widget logic to control widget display*
- *Displaying widgets inside of posts and pages*

Adding widget areas to your themes

Widgets are useful and can help you to achieve a lot of tasks on your site. Having a choice of widget areas in your theme means you have options. Sometimes you find, however, that your theme needs a widget area where one does not exist. It's an easy problem to resolve; all you need to do is add a bit of code to your theme and you can add the additional widget areas you desire.

> Many WordPress and third party documents refer to widget areas as 'dynamic sidebars.' These two names mean the same thing in this context.

Getting ready

To complete the following recipe, all that you need is a code editor and access to the theme files of your WordPress installation.

How to do it...

Let's add a new widget area to the right sidebar of the default TwentyTen theme. To accomplish this you will need to accomplish two tasks: first, register the new widget area, and then place it in the theme files.

1. Open your theme's `functions.php` (or create a new one if your theme does not include one) and let's register a new widget area called `tailormade`. To do this, add the following lines of code to the file:

```
if ( function_exists('register_sidebar') ){
   register_sidebar(array(
      'name' => 'tailormade',
      'before_widget' => '<div id="my-mega-menu-widget">',
      'after_widget' => '</div>',
      'before_title' => '<h3 class="widget-title">',
      'after_title' => '</h3>',
));

}
```

2. Save the file.
3. Open your theme's `sidebar.php` file and add the following code where you want the new widget area to appear:

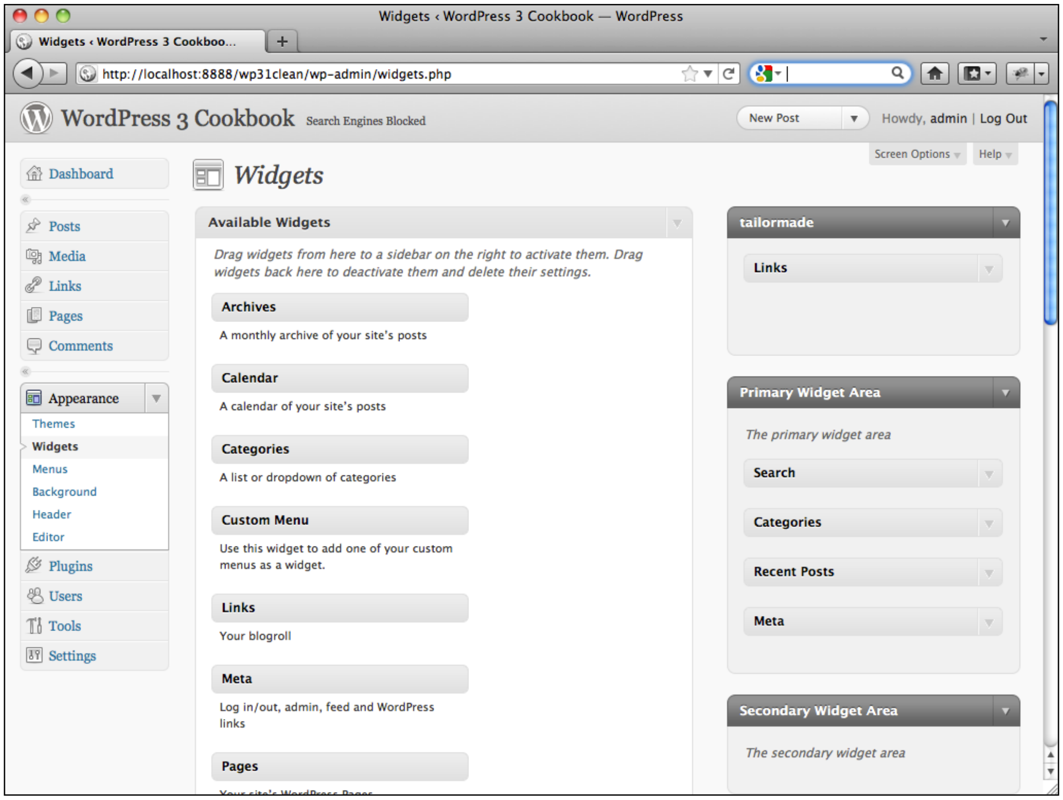
```
<?php if ( !function_exists('dynamic_sidebar') dynamic_
sidebar('tailormade') ) : ?> <?php endif; ?>
```

> Though it is not absolutely necessary, WordPress coding standards prefer widgets to be placed inside of lists. Best practices would dictate that you make your new widget area a list item `<li>` of an unordered list `<ul>`. Also, try not to hard code unnecessary styling, as it may have undesired results when it interacts with the widget's styling.

4. Save the file.

The preceding example is basic, but the principles demonstrated are applicable to other scenarios. You can use the technique to add multiple widget areas anywhere in your theme's templates.

As a result of this code, you now have a new widget area available on the dashboard's widgets page, as shown in the next screenshot. You will also have widget output visible on the front-end of the site.


Widgets ‹ WordPress 3 Cookbook — WordPress
Widgets ‹ WordPress 3 Cookboo...
http://localhost:8888/wp31clean/wp-admin/widgets.php
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Background
Header
Editor
Plugins
Users
Tools
Settings
Widgets
Available Widgets
Archives
Calendar
Categories
Custom Menu
Links
Meta
Pages
tailormade
Links
Primary Widget Area
Search
Categories
Recent Posts
Meta
Secondary Widget Area

Drag widgets from here to a sidebar on the right to activate them. Drag widgets back here to deactivate them and delete their settings.

A monthly archive of your site's posts

A calendar of your site's posts

A list or dropdown of categories

Use this widget to add one of your custom menus as a widget.

Your blogroll

Log in/out, admin, feed and WordPress links

The primary widget area

The secondary widget area

How it works...

When we were adding a single widget-ready zone, we used the `dynamic_sidebar()` function. This is why I used `dynamic_sidebar('tailormade')` in the sidebar template. Then, in the `functions.php` file, I used the `register_sidebar()` function, which can only handle a single widget-ready zone.

> To learn more about the `register_sidebar()` function and the parameters available, see the WordPress Codex at `http://codex.wordpress.org/Function_Reference/register_sidebar`

See also

- *Installing widgets*
- *Using widget logic to control widget display*
- *Displaying widgets inside of posts and pages*

Creating your own widget

Even though there are a lot of quality WordPress widgets available, sometimes you will find that there isn't one that exactly fits the requirements. In such a case, you'll have to create your own widget. It may sound difficult at first, but if you have any programming experience it is very easy.

Creating a widget requires creating the plugin that powers the widget; this chapter is, therefore, not about simply creating a widget, but also about creating a plugin. We will create a plugin, which in turn provides us with a widget.

Getting ready

To create your very own widget, you need nothing but a text editor and access to the WordPress installation on your server.

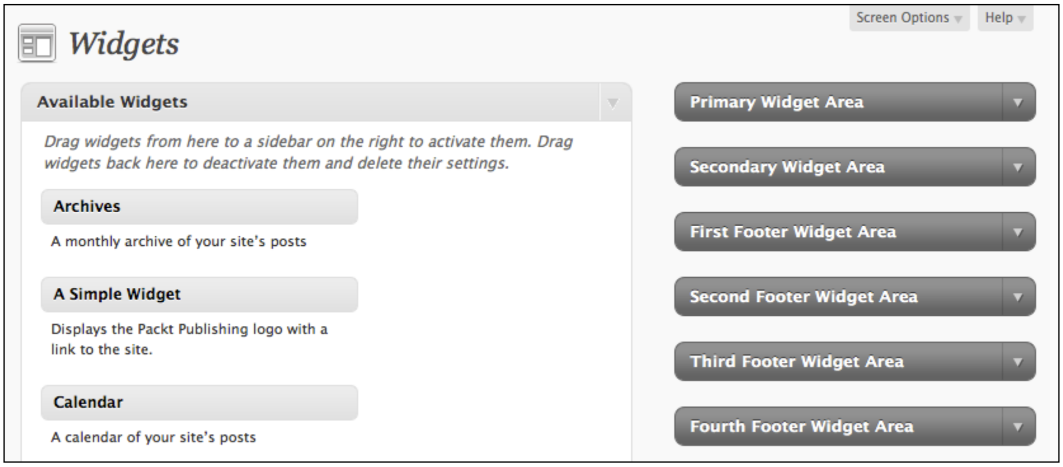
How to do it...

1. Find the directory `/wp-content/plugins`.
2. Create a new directory inside and name it `simplewidget`.
3. Create a blank file and name `simplewidget.php`.
4. Save it to the `wp-content/plugins/simplewidget` directory.
5. Open the `simplewidget.php` file for editing.
6. Paste into `simplewidget.php` the following code:

```
<?php
/*
Plugin Name: A Simple Widget Example

Plugin URI: http://yoursite.com/

Version: 1.0

Description: Demonstrate how to create a simple widget.

Author: The Great Gazoo

Author URI: http://yoursite.com/
*/

/**Creates new class to extend the system's WP_Widget class*/

class simple_widget extends WP_Widget {
   function simple_widget() {
      $widget_ops = array('description' => 'Displays the Packt
Publishing logo with a link to the site.');
      $this->WP_Widget('simple_widget', 'A Simple Widget',
$widget_ops);
   }

/**Implement the form() function, which creates the widget's
editing options for admin.*/

   function form($instance) {
      $instance = wp_parse_args((array) $instance, array( 'title'
=> '' ));
      $title = $instance['title'];
?>
   <p>
   <label for="<?php echo $this->get_field_id('title'); ?>"><?php
_e('Title:'); ?></label>
   <input class="widefat" id="<?php echo $this->get_field_
id('title'); ?>" name="<?php echo $this->get_field_name('title');
?>" type="text" value="<?php echo attribute_escape($title); ?>" />
   </p>
   <?php
   }

/**Implements the update() function, which allows us to set
options for the widget*/

   function update($new_instance, $old_instance) {

      $instance = $old_instance;
      $instance['title'] = $new_instance['title'];
      return $instance;
   }

/**Implements the widget() function, where we build the widget
output.*/

   function widget($args, $instance) {
      extract($args, EXTR_SKIP);

      echo $before_widget;
      $title = empty($instance['title']) ? '' : apply_
filters('widget_title', $instance['title']);
```

```
        if (!empty($title))
           echo $before_title . $title . $after_title;;
           echo '<a href="http://www.packtpub.com"><img src="http://
www.packtpub.com/sites/all/themes/pixture_reloaded/images/pp/
packt-logo.jpg"></a>';

        echo $after_widget;
    }
}

/**Time to register the widget*/

add_action( 'widgets_init', create_function('', 'return register_
widget("simple_widget");') );

?>
```

The plugin will now appear in your Plugins Manager. Activate and you will find a new widget in your list of widgets, seen as follows.


Screen Options
Help
Widgets
Available Widgets

Drag widgets from here to a sidebar on the right to activate them. Drag widgets back here to deactivate them and delete their settings.

Archives

A monthly archive of your site's posts

A Simple Widget

Displays the Packt Publishing logo with a link to the site.

Calendar

A calendar of your site's posts

Primary Widget Area
Secondary Widget Area
First Footer Widget Area
Second Footer Widget Area
Third Footer Widget Area
Fourth Footer Widget Area

Once you place the widget into a visible widget area, you will be able to see your first custom widget on your site.

How it works...

Now that we have created a working widget, let's take a function-by-function look at its code.

First, you have to tell WordPress that this file is a widget or a plugin by adding the PHP comments at the beginning of the file. The information will be used by your WordPress dashboard when you activate the widget. If WordPress can't find these comments, the file won't be recognized.

The first actual code begins with declaring our new class `simple_widget` and specifying that it is an extension of the WordPress system's `WP_Widget`. This first block of code also includes the function `simple_widget`, where we specify the description and name seen in the admin system.

The next function is `form()`. As the name implies, this function sets up any configuration options that will be available to the admin in the Widgets Manager. In this case, we have only given the administrator the option to specify a title for the widget.

The `update()` function is needed to enable the admin to make changes to the widget and then see those reflected in the widget output. In this case, the only option needed relates to the title.

Next comes the `widget()` function where we build the widget output for the front-end of the site. In this case, we tell the function to print the title, followed by an image file which has a link attached to it.

The last bIt of code will register our widget and add it to the admin interface; the user will be able to drag and drop it into one of his widget-ready zones.

Modifying core widgets

In the previous recipes, we showed you how to create new widgets and how to add new widget areas to your theme. As you know, WordPress comes with a number of built-in widgets including the categories list, the search form, and so on. While many of the default widgets are useful, they are also often lacking in any meaningful configuration options. If you want one of the default widgets to look or function differently, you will need to either find another solution or modify the widget.

In this recipe, we'll show you how to modify WordPress core widgets – without editing the core files.

> While it is possible to change your site's core widgets by directly editing the files that produce them, this is not recommended. The better choice is to create copies of them and make your modifications there. If you modify the core files, you run the risk of over-writing your changes when you update or upgrade your site.

Getting ready

To execute this recipe, you will need a code editor and access to the files of your WordPress site.

How to do it...

Let's say that you'd like to use the WordPress Meta widget, but you'd like to remove the link to WordPress.org.

1. Access your WordPress installation on your server.
2. Find and open the file `wp-includes/default-widgets.php`.
3. Search through the code in that file and find the `Meta widget` class, which appears as follows:

```
/**

* Meta widget class
*

* Displays log in/out, RSS feed links, etc.
*
* @since 2.8.0
*/
class WP_Widget_Meta extends WP_Widget {
   function WP_Widget_Meta() {
      $widget_ops = array('classname' => 'widget_meta',
'description' => __( "Log in/out, admin, feed and WordPress
links") );
      $this->WP_Widget('meta', __('Meta'), $widget_ops);
   }

   function widget( $args, $instance ) {
      extract($args);
      $title = apply_filters('widget_title',
empty($instance['title']) ? __('Meta') : $instance['title'],
$instance, $this->id_base);

      echo $before_widget;
      if ( $title )
         echo $before_title . $title . $after_title;
?>

         <ul>

         <?php wp_register(); ?>
         <li><?php wp_loginout(); ?></li>

         <li><a href="<?php bloginfo('rss2_url'); ?>" title="<?php
echo esc_attr(__('Syndicate this site using RSS 2.0')); ?>"><?php
_e('Entries <abbr title="Really Simple Syndication">RSS</abbr>');
?></a></li>

         <li><a href="<?php bloginfo('comments_rss2_url'); ?>"
title="<?php echo esc_attr(__('The latest comments to all posts
in RSS')); ?>"><?php _e('Comments <abbr title="Really Simple
Syndication">RSS</abbr>'); ?></a></li>

         <li><a href="http://wordpress.org/" title="<?php echo
esc_attr(__('Powered by WordPress, state-of-the-art semantic
personal publishing platform.')); ?>">WordPress.org</a></li>

         <?php wp_meta(); ?>
         </ul>
<?php
      echo $after_widget;
   }

   function update( $new_instance, $old_instance ) {
      $instance = $old_instance;
      $instance['title'] = strip_tags($new_instance['title']);

      return $instance;
   }

   function form( $instance ) {
      $instance = wp_parse_args( (array) $instance, array( 'title'
=> '' ) );
      $title = strip_tags($instance['title']);
?>
         <p><label for="<?php echo $this->get_field_id('title');
?>"><?php _e('Title:'); ?></label> <input class="widefat"
id="<?php echo $this->get_field_id('title'); ?>" name="<?php echo
$this->get_field_name('title'); ?>" type="text" value="<?php echo
esc_attr($title); ?>" /></p>

<?php
   }
}
```

4. Copy the `Meta widget` code.
5. Create a new directory inside `wp-content/plugins`. Name it `metamod`.

6. Create a new file inside the `metamod` directory and name it `metamod.php`.
7. Paste the `Meta widget` code into the `metamod.php` file.
8. We're going to modify the code to accomplish several tasks: We're going to give the function a unique name, include a proper plugin header, make our changes to exclude the wordpress.org link, and finally register the new widget. The modified code will look like this; the changes are explained in the text after the code:

```
<?php
/*
Plugin Name: Meta Mod

Plugin URI: http://yoursite.com/

Version: 1.0

Description: Holds modified version of the Meta widget.

Author: The Great Gazoo

Author URI: http://yoursite.com/
*/



class Meta_Mod extends WP_Widget {

   function Meta_Mod() {
      $widget_ops = array('classname' => 'widget_meta',
'description' => __( "Log in/out, admin, feed and WordPress
links") );
      $this->WP_Widget('meta', __('Meta'), $widget_ops);
   }

   function widget( $args, $instance ) {
      extract($args);
      $title = apply_filters('widget_title',
empty($instance['title']) ? __('Meta') : $instance['title'],
$instance, $this->id_base);

      echo $before_widget;
      if ( $title )
         echo $before_title . $title . $after_title;
?>

         <ul>
         <?php wp_register(); ?>
         <li><?php wp_loginout(); ?></li>

         <li><a href="<?php bloginfo('rss2_url'); ?>" title="<?php
echo esc_attr(__('Syndicate this site using RSS 2.0')); ?>"><?php
_e('Entries <abbr title="Really Simple Syndication">RSS</abbr>');
?></a></li>

         <li><a href="<?php bloginfo('comments_rss2_url'); ?>"
title="<?php echo esc_attr(__('The latest comments to all posts
in RSS')); ?>"><?php _e('Comments <abbr title="Really Simple
Syndication">RSS</abbr>'); ?></a></li>
         </ul>
<?php
      echo $after_widget;
   }

   function update( $new_instance, $old_instance ) {
      $instance = $old_instance;
      $instance['title'] = strip_tags($new_instance['title']);

      return $instance;
   }

   function form( $instance ) {
      $instance = wp_parse_args( (array) $instance, array( 'title'
=> '' ) );
      $title = strip_tags($instance['title']);
?>
         <p><label for="<?php echo $this->get_field_id('title');
?>"><?php _e('Title:'); ?></label> <input class="widefat"
id="<?php echo $this->get_field_id('title'); ?>" name="<?php echo
$this->get_field_name('title'); ?>" type="text" value="<?php echo
esc_attr($title); ?>" /></p>

<?php
   }
}

/**Time to register the widget*/

add_action( 'widgets_init', create_function('', 'return register_
widget("Meta_Mod");') );

?>
```

How it works...

The idea behind this recipe is to take the code from a default widget, use it as the basis of a new plugin, which will then provide us with a widget we can modify as we wish. Once installed, we will use the new widget and remove the default widget from display on the site.

The process is very straightforward. The vast majority of the work is simply copying the existing widget code and pasting it into a new file. We have to change the name of the function so that it does not conflict with the original function. To make that change, we simply modify the beginning of the code to make this change:

```
class WP_Widget_Meta extends WP_Widget {


   function WP_Widget_Meta() {
```

It becomes the following:

```
class Meta_Mod extends WP_Widget {


   function Meta_Mod() {
```

As we saw in the previous recipe, creating a widget requires a proper header, so we add the following to the top of the file:

```
<?php
/*
Plugin Name: Meta Mod

Plugin URI: http://yoursite.com/

Version: 1.0

Description: Holds modified version of the Meta widget.

Author: The Great Gazoo

Author URI: http://yoursite.com/
*/
```

Finally, to make the plugin work, we need to register our new function. To accomplish this, we added the following to the end of the file:

```
/**Time to register the widget*/

add_action( 'widgets_init', create_function('', 'return register_
widget("Meta_Mod");') );

?>
```

That's all there is to make the plugin and widget work, but our original goal was to delete the link to WordPress.org, so we also need to edit a portion of the code, as follows:

```
         <ul>

         <?php wp_register(); ?>
         <li><?php wp_loginout(); ?></li>

         <li><a href="<?php bloginfo('rss2_url'); ?>" title="<?php
echo esc_attr(__('Syndicate this site using RSS 2.0')); ?>"><?php
_e('Entries <abbr title="Really Simple Syndication">RSS</abbr>'); ?></
a></li>

         <li><a href="<?php bloginfo('comments_rss2_url'); ?>"
title="<?php echo esc_attr(__('The latest comments to all posts
in RSS')); ?>"><?php _e('Comments <abbr title="Really Simple
Syndication">RSS</abbr>'); ?></a></li>

         <li><a href="http://wordpress.org/" title="<?php echo esc_
attr(__('Powered by WordPress, state-of-the-art semantic personal
publishing platform.')); ?>">WordPress.org</a></li>

         <?php wp_meta(); ?>
         </ul>
```

This becomes the following:

```
<ul>

         <?php wp_register(); ?>
         <li><?php wp_loginout(); ?></li>

         <li><a href="<?php bloginfo('rss2_url'); ?>" title="<?php
echo esc_attr(__('Syndicate this site using RSS 2.0')); ?>"><?php
_e('Entries <abbr title="Really Simple Syndication">RSS</abbr>'); ?></
a></li>
```

```
            <li><a href="<?php bloginfo('comments_rss2_url'); ?>"
title="<?php echo esc_attr(__('The latest comments to all posts
in RSS')); ?>"><?php _e('Comments <abbr title="Really Simple
Syndication">RSS</abbr>'); ?></a></li>
            </ul>
```

Once these changes are done, you will find a new plugin on your site. Once you activate it, you will have a new widget named Meta Mod.

> You will also still have the original Meta widget, which has not been changed.

Displaying tabs on your sidebar

If you are like many site owners, there are a number of items you would like to display in your widget areas. The problem is, if you try to put too many items into a single widget area, the area can become quite lengthy or crowded, which isn't visually appealing.

In this recipe we look at one possible solution for this problem: Implementing tabbed widgets.

Getting ready

To execute this recipe, you will need to install the Tabbed Widgets plugin. You will need to install this plugin before you can get started. Search for **Tabbed Widget** inside the **Add New Plugins** screen of your WordPress site. After you find it, click to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://konstruktors.com/projects/wordpress-plugins/tabbed-accordion-widgets/`

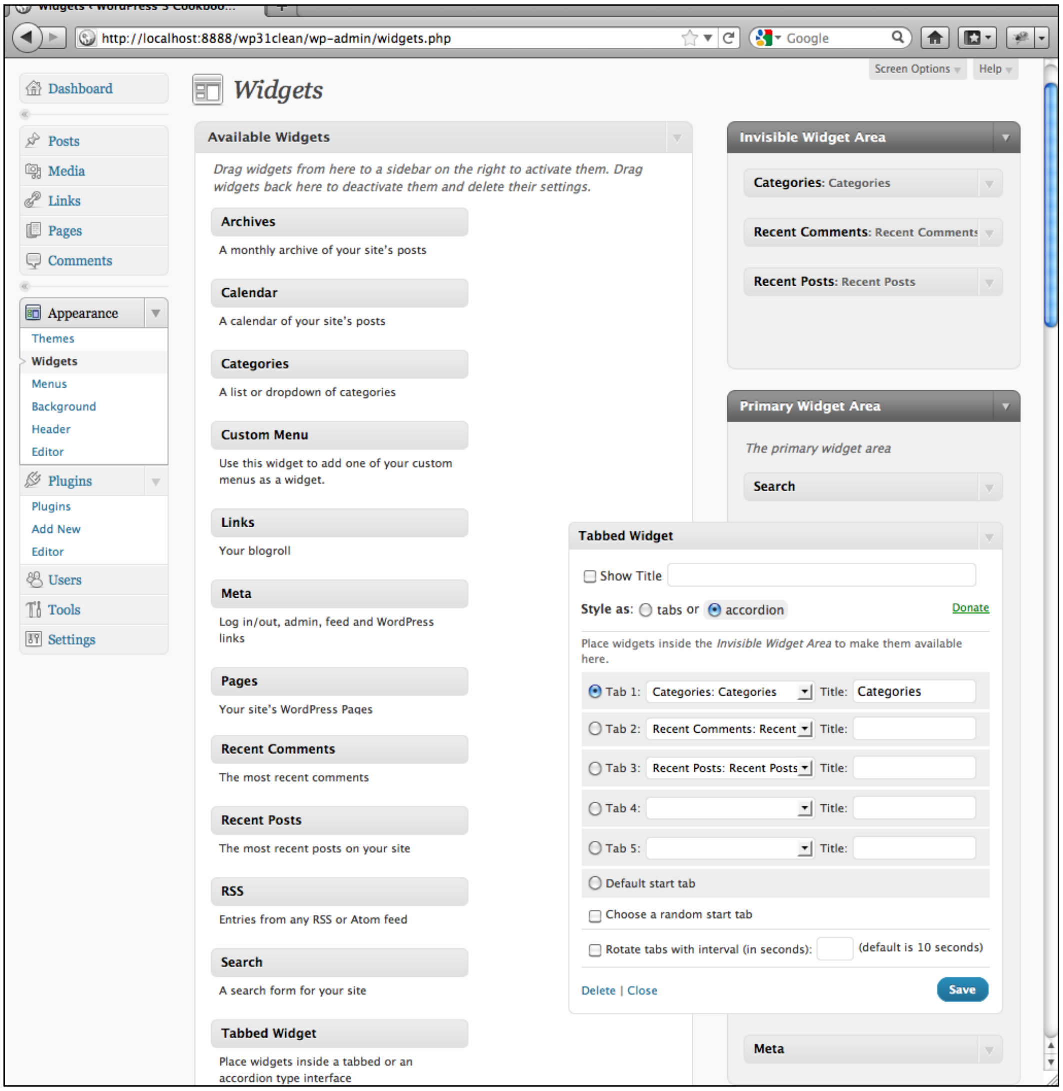
How to do it...

Though Tabbed Widgets is a plugin, once you activate it, all other work occurs in the Widgets Manager. Widget placement and configuration works a bit differently than usual, expalined as follows:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.
3. Select the option **Widgets**.
4. Click and drag the **Tabbed Widget** widget to the widget area where you wish it to appear.

> There is a new widget area named **Invisible Widget Area**. Drag into that widget area any widgets you want to appear in the new Tabbed Widget.

5. Next, open the **Tabbed Widget**. The following screenshot shows what it will look like:


http://localhost:8888/wp31clean/wp-admin/widgets.php
Google
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Background
Header
Editor
Plugins
Plugins
Add New
Editor
Users
Tools
Settings
Widgets
Available Widgets
Drag widgets from here to a sidebar on the right to activate them. Drag widgets back here to deactivate them and delete their settings.
Archives
A monthly archive of your site's posts
Calendar
A calendar of your site's posts
Categories
A list or dropdown of categories
Custom Menu
Use this widget to add one of your custom menus as a widget.
Links
Your blogroll
Meta
Log in/out, admin, feed and WordPress links
Pages
Your site's WordPress Pages
Recent Comments
The most recent comments
Recent Posts
The most recent posts on your site
RSS
Entries from any RSS or Atom feed
Search
A search form for your site
Tabbed Widget
Place widgets inside a tabbed or an accordion type interface
Invisible Widget Area
Categories: Categories
Recent Comments: Recent Comments
Recent Posts: Recent Posts
Primary Widget Area
The primary widget area
Search
Tabbed Widget
Donate
Place widgets inside the Invisible Widget Area to make them available here.
Delete | Close
Save
Meta

☐ Show Title

Style as: ○ tabs or ● accordion

● Tab 1: Categories: Categories Title: Categories

○ Tab 2: Recent Comments: Recent Title:

○ Tab 3: Recent Posts: Recent Posts Title:

○ Tab 4: Title:

○ Tab 5: Title:

○ Default start tab

☐ Choose a random start tab

☐ Rotate tabs with interval (in seconds): (default is 10 seconds)

6. Select what you want to appear in each tab.
7. Select the style of the appearance, either tabs or accordion style.

8. Select any other configuration options you want to use.
9. Click on the **Save** button.

How it works...

The plugin enables you to stack multiple widgets into the space taken by one. The configuration options allow you to then either display the widgets in a set of horizontal tabs or in a set of vertical accordion menus. The plugin has very little styling, so you will most likely want to style this yourself with a bit of CSS.

Using conditional tags to control widget display

While WordPress widgets are a great way to create pages that are rich in functionality and content, there are times when you want to be able to control which widgets show on which pages. By default, a widget placed in a widget area will appear on all pages of your site. Some themes, like the default TwentyTen, provide the option to show one set of widgets on the home page and a different set on the interior pages, but sometimes you want more variety.

In this recipe, we explore implementing the Widget Logic plugin, which enables the use of WordPress conditional tags in widgets through the Widgets Manager.

> Conditional tags are typically used in template files to control the display of items depending on what conditions that page matches. You can learn more about conditional tags on the WordPress Codex at `http://codex.wordpress.org/Conditional_Tags`

Getting ready

To execute this recipe, you will need to install the Widget Logic plugin. You will need to install this plugin before you can get started. Search for **Widget logic** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

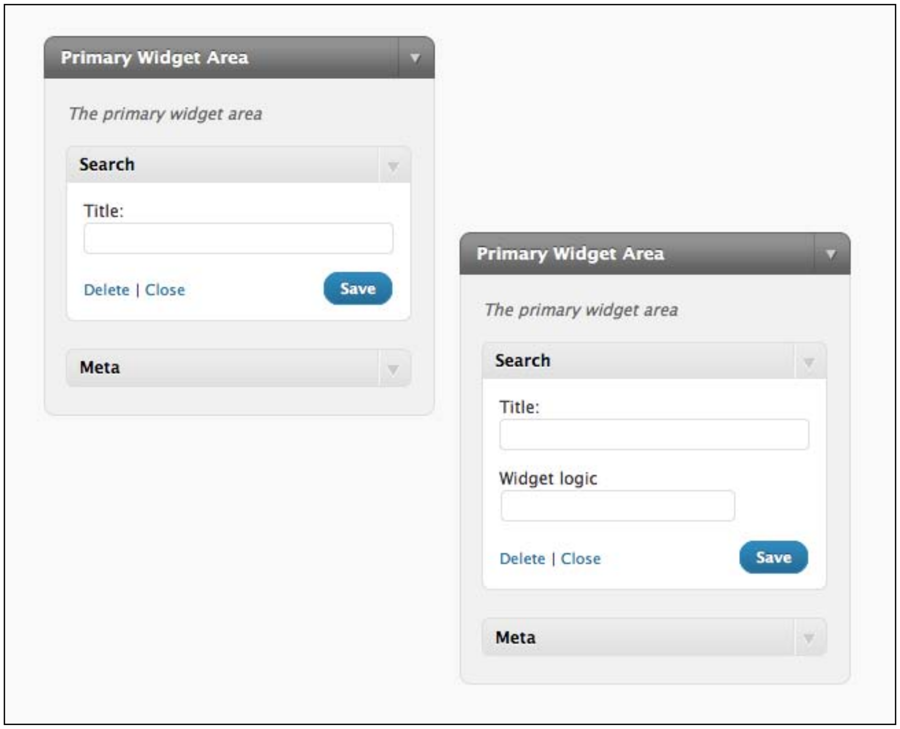
> You can learn more about the plugin by visiting the developer's website at `http://freakytrigger.co.uk/wordpress-setup/`

How to do it...

There are no configuration options associated with this plugin. Once it is installed and activated, it is ready to use. The plugin works by adding a new field to all the widgets on your site. That field is labelled simply **Widget logic** and you will use this field to inset the conditional tags that control that particular widget. The screenshot below shows the appearance of the **Search** widget before and after the Widget Logic plugin is activated; note the **Widget logic** field in the second example.

By way of example, assume that you want to only display the Meta widget on the home page of your site. To implement this using the Widget Logic plugin, you would follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.
3. Click the **Widgets** option.
4. Find the **Meta** widget (by default it is assigned to the Primary Widget Area).
5. Open the options for the Meta widget.
6. In the **Widget logic** field, type the following conditional tag: `is_front_page()`
7. Click on the **Save** button.


Primary Widget Area
The primary widget area
Search
Title:
Delete | Close
Save
Meta
Primary Widget Area
The primary widget area
Search
Title:
Widget logic
Delete | Close
Save
Meta

How it works...

The Widget Logic plugin simply adds the power of the WordPress conditional tags functionality to your widgets. The entire functionality hinges on conditional tags, which are part of the WordPress API. To use this plugin effectively, you do need to understand the conditional tags options available to you and the proper syntax.

There's more...

In the context of widget display, some of the most useful conditional tags are:

- `is_author()`: Only displays widget to visitors viewing an Author page. This tag includes variables that also allow you to name specific authors.
- `in_category()`: Only displays widget on posts within a specific Category.
- `is_category()`: Only displays widget on Category archive pages. There are variables available which allow you to restrict display to specific categories.
- `is_front_page()`: Only displays widget on the front page of the website.
- `is_page()`: Displays widget when any page is being viewed. Variables allow more specificity.
- `is_page_template()`: Display widget when a specific page template is being used.
- `is_single()`: Display when any single post, or custom Post Type, is shown.
- `is_sticky()`: Display widget when a post is marked as sticky.
- `is_tag()`: Display widget when any tag archive page is displayed. Variables allow you to tailor the display to specific pages.
- `has_tag()`: Display widget when the post has a specific tag. Variables allow more specificity.

> To learn more about conditional tags, visit the WordPress Codex at `http://codex.wordpress.org/Conditional_Tags`

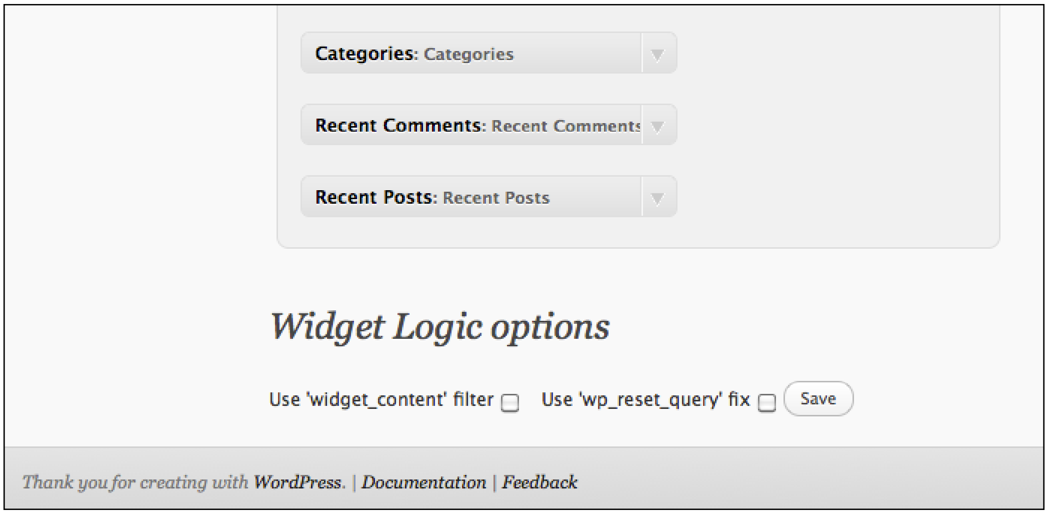
Using the widget content filter

An additional functionality of the plugin is hidden away down on the bottom of the Widgets page. Note the screenshot below, which shows the options in their default state.

The **widget_content filter** is a more advanced functionality. If you enable the **widget_content filter** you can modify the text displayed by any widget on your site by hooking into the filter in your theme's `function.php` file. A typical usage would be to add additional styling to your widgets, thereby giving you more control over their appearance or placement.

The **widget_content filter** is discussed more on the plugin's page on wordpress.org. Visit: `http://wordpress.org/extend/plugins/widget-logic/other_notes/`

The second checkbox you see in the screenshot below, **Use 'wp_reset_query fix'**, should only be used if you are experiencing display problems with widgets employing widget logic:



Categories: Categories
Recent Comments: Recent Comments
Recent Posts: Recent Posts

Widget Logic options

Use 'widget_content' filter ☐ Use 'wp_reset_query' fix ☐ Save

Thank you for creating with WordPress. | Documentation | Feedback

See also

- *Installing widgets*
- *Adding widget areas to your theme*
- *Displaying widgets inside of posts and pages*

Displaying widgets inside of posts and pages

As we saw earlier in this chapter, widget placement options on your WordPress site are a by-product of your theme. The options you have are a direct result of what the theme developer has provided for you. Wouldn't it be nice if you could put widgets wherever you like, without having to edit your theme files? This recipe addresses part of that need, by opening up your content area for widget placement.

In this recipe, we look at implementing a plugin that enables you to place widgets directly inside of your posts and pages.

Getting ready

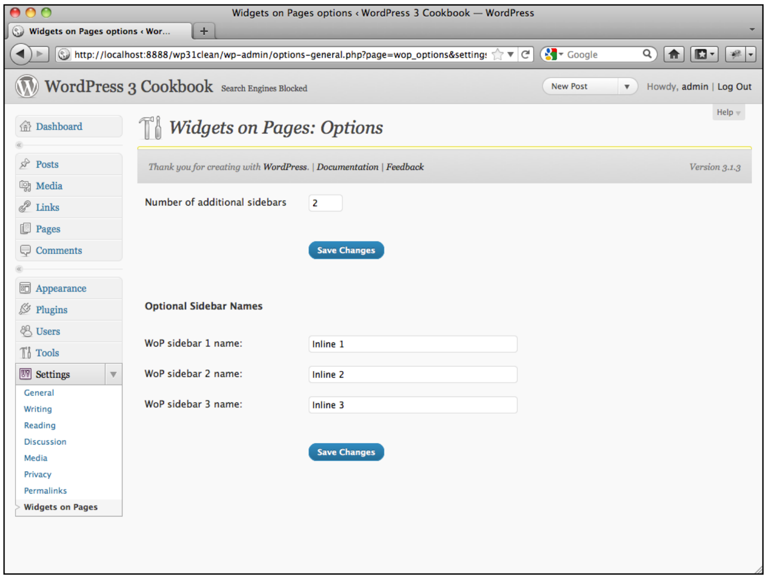
To execute this recipe, you will need to install the Widgets on Pages plugin. You will need to install this plugin before you can get started. Search for **Widgets on Pages** inside the **Add New Plugins** screen of your WordPress site. After you find it, click to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://gingerbreaddesign.co.uk/wordpress/widgets-on-pages/`

How to do it...

Before you can begin to insert widgets into your pages and posts, you need to first create the extra widget areas you want, then assign widgets to those areas. Follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the option **Widgets on Pages**.
4. Select how many additional widget areas (what the plugin calls "sidebars") you need by entering a value in the field labeled **Number of additional sidebars**.
5. Click on **Save Changes**.
6. In the fields provided, give names to each of your new widget areas, as shown in the following screenshot:



Widgets on Pages options ‹ WordPress 3 Cookbook — WordPress

http://localhost:8888/wp31clean/wp-admin/options-general.php?page=wop_options&setting:

Widgets on Pages: Options

Thank you for creating with *WordPress*. | *Documentation* | *Feedback* *Version 3.1.3*

Number of additional sidebars: 2

Optional Sidebar Names

WoP sidebar 1 name: Inline 1

WoP sidebar 2 name: Inline 2

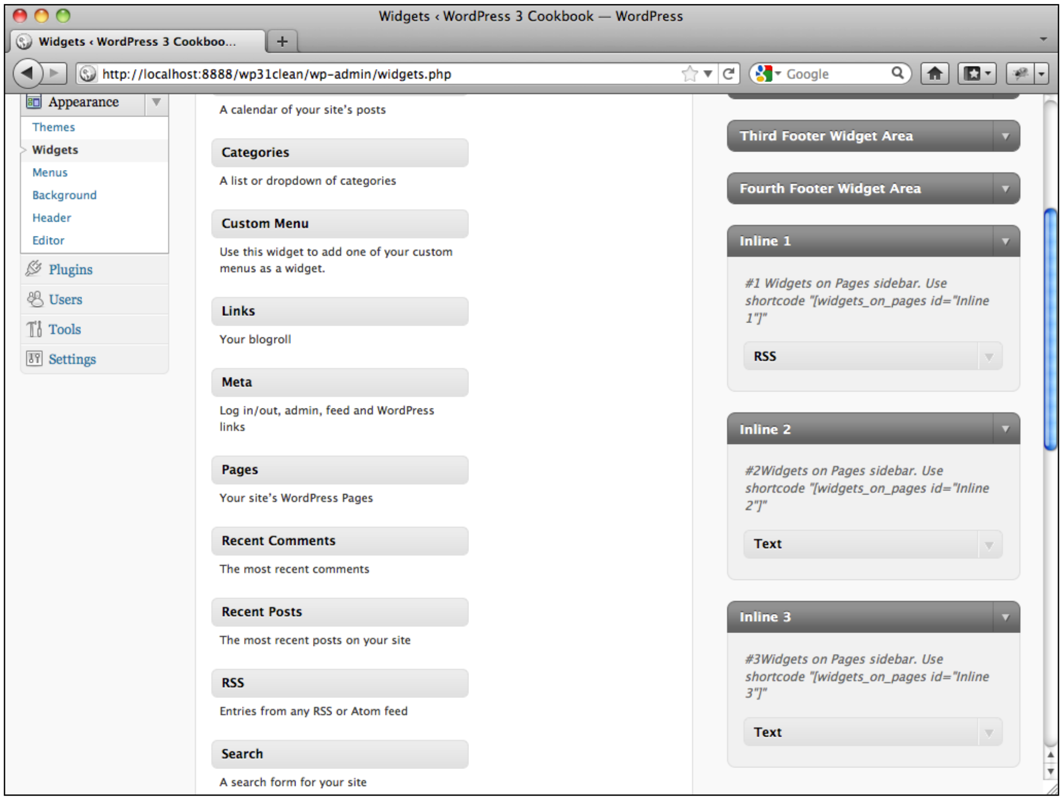
WoP sidebar 3 name: Inline 3

Widgets on Pages options ‹ Wor...
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Widgets on Pages
Save Changes
Save Changes

7. Click on the **Save Changes** button.

Next, let's assign widgets to the new areas:

1. Click on the **Appearance** menu.
2. Click on the option **Widgets**.
3. You will see the new widget areas you just created on the bottom right. Click and drag the widgets you want to use into the new widget areas, as shown in the following screenshot:


Widgets ‹ WordPress 3 Cookbook — WordPress
Widgets ‹ WordPress 3 Cookboo...
http://localhost:8888/wp31clean/wp-admin/widgets.php
Google
Appearance
Themes
Widgets
Menus
Background
Header
Editor
Plugins
Users
Tools
Settings
Categories
Custom Menu
Links
Meta
Pages
Recent Comments
Recent Posts
RSS
Search
Third Footer Widget Area
Fourth Footer Widget Area
Inline 1
RSS
Inline 2
Text
Inline 3
Text

A calendar of your site's posts

A list or dropdown of categories

Use this widget to add one of your custom menus as a widget.

Your blogroll

Log in/out, admin, feed and WordPress links

Your site's WordPress Pages

The most recent comments

The most recent posts on your site

Entries from any RSS or Atom feed

A search form for your site

#1 Widgets on Pages sidebar. Use shortcode "[widgets_on_pages id="Inline 1"]"

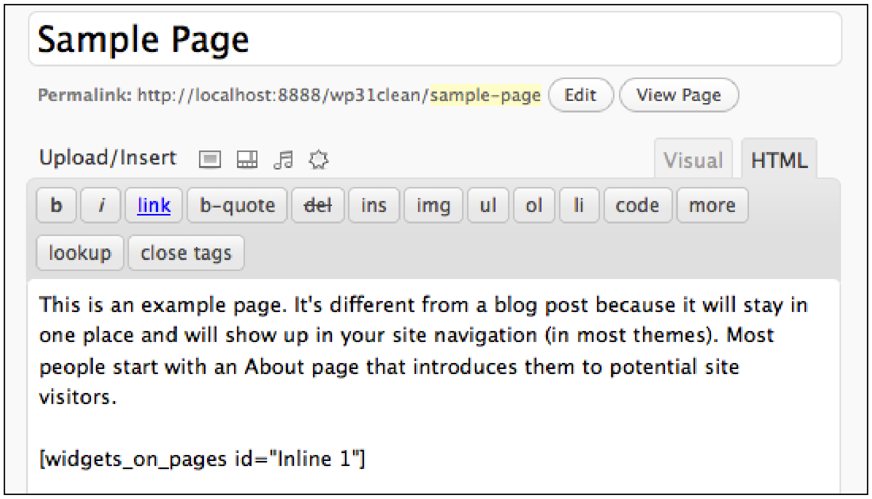
#2Widgets on Pages sidebar. Use shortcode "[widgets_on_pages id="Inline 2"]"

#3Widgets on Pages sidebar. Use shortcode "[widgets_on_pages id="Inline 3"]"

You are now set to add widgets to your pages or posts. To do so, you will insert WordPress shortcodes into each page or post. Let's assume you want to add an RSS feed to the content area of a specific page. We put the RSS widget into the Inline 1 widget area, above. Now, to get that into the page, follow these steps:

1. Open the page for editing.
2. Switch to the **HTML** editor.

3. Enter the shortcode provided where you want the output to appear. As you can see in the preceding screenshot (view the text in the widget area), the shortcode you will need to use is `[widgets_on_pages id="Inline 1"]`. The following screenshot shows how this would work:



Sample Page
Permalink: http://localhost:8888/wp31clean/sample-page
Edit
View Page
Upload/Insert
Visual
HTML
b
i
link
b-quote
del
ins
img
ul
ol
li
code
more
lookup
close tags

This is an example page. It's different from a blog post because it will stay in one place and will show up in your site navigation (in most themes). Most people start with an About page that introduces them to potential site visitors.

[widgets_on_pages id="Inline 1"]

4. Click the **Update** button to save the page.

How it works...

This plugin allows for the ad hoc creation of new widget areas, which the plugin refers to as sidebars. The plugin also creates a specific shortcut for each of the new widget areas. By placing the shortcode inside of pages or posts, you are able to place the custom widget area where you wish.

See also

- *Installing widgets*
- *Adding widget areas to your theme*
- *Using widget logic to control widget display*

4

Customizing Content Display

In this chapter, we will cover:

- How to work with the WordPress loop
- How you can retrieve posts from a specific category
- How to control how many posts you display
- Retrieving posts by date
- Showing only those posts published today
- How to show posts published exactly one year ago
- How to set up and use multiple loops
- How to access your post data from outside the WordPress loop
- How to access permalink information outside the loop
- How to display thumbnails on your posts, using custom fields
- How to implement alternating backgrounds for lists of posts
- How to display posts in a two column layout
- Creating WordPress shortcodes
- Enabling shortcode usage within widgets
- How to add notes to your posts
- How to enable tagging support for your site's page

Introduction

What is the WordPress loop? The loop is a group of PHP instructions that retrieve posts from the database of your WordPress site and then displays them on the page. You can find the WordPress loop inside your themes, for example, inside `index.php`, `single.php`, or `page.php`. Any HTML or PHP code placed between the beginning of the loop and the end will be used for each post.

At its most basic, a simple implementation of the loop could work like this:

```
<?php if (have_posts()) : while (have_posts()) : the_post(); ?>
<?php the_title(); ?>
<?php the_content(); ?>
<?php endwhile; else: ?>
   <p>Sorry, no posts matched your criteria.'</p>
<?php endif; ?>
```

In the real world, however, the WordPress loop is rarely that simple. This is one of those concepts best explained by referring to a real world example, so open up the `index.php` file of your system's TwentyEleven theme. Look for the following lines of code:

```
<?php if ( have_posts() ) : ?>

   <?php twentyeleven_content_nav( 'nav-above' ); ?>

      <?php /* Start the Loop */ ?>
      <?php while ( have_posts() ) : the_post(); ?>

         <?php get_template_part( 'content', get_post_format()
); ?>

      <?php endwhile; ?>

   <?php twentyeleven_content_nav( 'nav-below' ); ?>

<?php else : ?>

   <article id="post-0" class="post no-results not-found">
      <header class="entry-header">
         <h1 class="entry-title"><?php _e( 'Nothing Found',
'twentyeleven' ); ?></h1>
      </header><!-- .entry-header -->

      <div class="entry-content">
```

```
         <p><?php _e( 'Apologies, but no results were found
for the requested archive. Perhaps searching will                help
find a related post.', 'twentyeleven' ); ?></p>
         <?php get_search_form(); ?>
      </div><!-- .entry-content -->
   </article><!-- #post-0 -->

<?php endif; ?>
```

Most of the extra stuff seen in the loop from TwentyEleven is there to add in additional page elements, including content navigation; there's also some code to control what happens if there are no posts to display. The nature of the WordPress loops means that theme authors can add in what they want to display and thereby customize and control the output of their site.

> As you would expect, the WordPress Codex includes an extensive discussion of the WordPress loop. Visit `http://codex.wordpress.org/The_Loop`.

Accessing posts within the WordPress loop

In this recipe, we look at how to create a custom template that includes your own implementation of the WordPress loop.

Getting ready

All you need to execute this recipe is your favorite code editor and access to the WordPress files on your server. You will also need a theme template file, which we will use to hold our modified WordPress loop.

How to do it

Let's assume you have created a custom template. Inside of that template you will want to include the WordPress loop. Follow these steps to add the loop, along with a little customization:

1. Access the active theme files on your WordPress installation.
2. Find a template file and open it for editing. If you're not sure which one to use, try the `index.php` file.
3. Add to the file the following line of code, which will start the loop:

   ```
   <?php if ( have_posts() ) : while ( have_posts() ) : the_post();
   ?>
   ```

4. Next, let's display the post title, wrapped in an `h2` tag:

   ```
   <h2><?php the_title() ?></h2>
   ```

5. Let's also add a link to all posts by this author. Add this code immediately below the previous line:

   ```
   <?php the_author_posts_link() ?>
   ```

6. For the post content, let's wrap it in a `div` for easy styling:

   ```
   <div class="thecontent">
      <?php the_content(); ?>
   </div>
   ```

7. Next, let's terminate the loop and add some code to display a message if there were no posts to display:

   ```
   <?php endwhile; else: ?>
   <p>Oops! There are no posts to display.</p>
   ```

8. Finally, let's put a complete stop to the loop by ending the if statement that began the code in step number 3, above:

   ```
   <?php endif; ?>
   ```

9. Save the file.

That's all there is to it. Your code should look like this:

```
<?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>
   <h2><?php the_title() ?></h2>
   <?php the_author_posts_link() ?>
   <div class="thecontent">
      <?php the_content(); ?>
   </div>
   <?php endwhile; else: ?>
      <p>Oops! There are no posts to display.</p>
<?php endif; ?>
```

How it works...

This basic piece of code first checks if there are posts in your site. If the answer is yes, the loop will repeat until every post title and their contents are displayed on the page. The post title is displayed using `the_title()`. The author's name and link are added with `the_author_posts_link()` function. The content is displayed with `the_content()` function and styled by the div named `thecontent`. Finally, if there are no posts to display, the code will display the message **Oops! There are no posts to display**.

There's more...

In the preceding code you saw the use of two template tags: `the_author_posts_link` and `the_content`. These are just two examples of the many template tags available in WordPress. The tags make your life easier by reducing an entire function to just a short phrase. You can find a full list of the template tags at: `http://codex.wordpress.org/Template_Tags`.

The template tags can be broken down into a number of categories:

- **General tags**: The tags in this category cover general page elements common to most templates
- **Author tags**: Tags related to author information
- **Bookmark tags**: The tag to list bookmarks
- **Category tags**: Category, tag, and item description-related
- **Comment tags**: Tags covering the comment elements
- **Link tags**: Tags for links and permalinks
- **Post tags**: Tags for posts, excerpts, titles, and attachments
- **Post Thumbnail tags**: Tags that relate to the post thumbnails
- **Navigation Menu tags**: Tags for the nav menu and menu tree

Retrieving posts from a specific category

There are times when you might wish to display only those posts that belong to a specific category, for example, perhaps you want to show only the featured posts. With a small modification to the WordPress loop, it's easy to grab only those posts you want to display.

In this recipe we introduce `query_posts()`,which can be used to control which posts are displayed by the loop.

Getting ready

To execute this recipe, you will need a code editor and access to the WordPress files on your server. You will also need a theme template file, which we will use to hold our modified WordPress loop. To keep this recipe short and to the point, we use the loop we created in the preceding recipe.

How to do it...

Let's create a situation where the loop shows only those posts assigned to the Featured category. To do this, you will need to work through two different processes.

First, you need to find the category ID number of the Featured category. To do this, follow these steps:

1. Log in to the **Dashboard** of your WordPress site.
2. Click on the **Posts** menu.
3. Click on the **Categories** option.
4. Click on the category named **Featured**.
5. Look at the address bar of your browser and you will notice that part of the string looks something like this: **&tag_ID=9**. On the site where we are working, the Featured category has the `ID` of `9`.

> Category IDs vary from site to site. The ID used in this recipe may not be the same as the ID for your site!

Next, we need to add a query to our loop that will extract only those posts that belong to the Featured category, that is, to those posts that belong to the category with the `ID` of `9`. Follow these steps:

1. Open the file that contains the loop. We'll use the same file we created in the preceding recipe.
2. Find the loop:

```
<?php if ( have_posts() ) : while ( have_posts() ) :
    the_post(); ?>
   <h2><?php the_title() ?></h2>
   <?php the_author_posts_link() ?>
   <div class="thecontent">
      <?php the_content(); ?>
   </div>
<?php endwhile; else: ?>
   <p>Oops! There are no posts to display.</p>
<?php endif; ?>
```

3. Add the following line of code immediately above the loop:

```
<?php query_posts($query_string.'&cat=9'); ?>
```

4. Save the file.

That's all there is to it. If you visit your site, you will now see that the page displays only the posts that belong to the category with the `ID` of `9`.

How it works...

The `query_posts()` function modifies the default loop. When used with the `cat` parameter, it allows you to specify one or more categories that you want to use as filters for the posts.

For example:

- `query_posts(&query_string.'&cat=5');`: Get posts from the category with ID 5 only
- `query_posts(&query_string.'&cat=5,6,9');`: Get posts from the category with IDs 5, 6, and 9
- `query_posts(&query_string.'&cat=-3');`: Get posts from all categories, except those from the category with ID 3

> For more information, visit the WordPRess Codex page on query posts: `http://codex.wordpress.org/Function_Reference/query_posts`

Getting a specific number of posts

If you'd like to control the number of posts displayed on the page, you can do so by using the `query_posts()` function we introduced in the previous recipe.

Getting ready

All you need to execute this recipe is your favorite code editor and access to the WordPress files on your server. You will also need a theme template file, which we will use to hold our modified WordPress loop. To keep this recipe short and to the point, we use the loop we created in the first recipe in this chapter.

How to do it...

Let's create a situation where the loop produces only two posts for the page. Follow these steps:

1. Open the file that contains the loop. We'll use the same file we created in the first recipe in this chapter.

2. Find the loop:

   ```
   <?php if ( have_posts() ) : while ( have_posts() ) :
        the_post(); ?>
      <h2><?php the_title() ?></h2>
      <?php the_author_posts_link() ?>
      <div class="thecontent">
         <?php the_content(); ?>
      </div>
   <?php endwhile; else: ?>
      <p>Oops! There are no posts to display.</p>
   <?php endif; ?>
   ```

3. Add the following line of code immediately above the loop:

   ```
   <?php query_posts($query_string.'showposts=2'); ?>
   ```

4. Save the file.

That's all there is to it. If you visit your site, you will now see that the page displays only two posts (by default, the two most recent posts).

How it works...

Just like in the previous example, we're using the powerful `query_posts()` function. This time, we use the `showposts` parameter that allows you to specify how many posts must be displayed.

> While this recipe shows you how to hard code the number of posts, you can also control the number of posts per page from within the WordPress dashboard. Accordingly, this recipe should be used only in special circumstances, where the default controls will not meet your needs.

Retrieving posts by date

Another very handy use of the `query_posts()` function is to get posts according to specific date and time parameters. Retrieving posts by a specific date can be a bit more complex than, for example, getting an exact number of posts, because you sometimes need to use multiple parameters with the `query_posts()` function. The example in this recipe, however, is rather simple. We'll see more complex uses of `query_posts()` date and time parameters in the next recipes.

Let's see how to use `query_posts()` to retrieve and display the posts from a specific date.

Getting ready

All you need to execute this recipe is your favorite code editor and access to the WordPress files on your server. You will also need a theme template file, which we will use to hold our modified WordPress loop. To keep this recipe short and to the point, we use the loop we created in the first recipe in this chapter.

How to do it...

Let's create a situation where the loop produces only the posts that were published in October. Follow these steps:

1. Open the template file that contains the loop. We'll use the same file we created in the first recipe in this chapter.
2. Find the loop:

```
         <?php if ( have_posts() ) : while ( have_posts() ) :
the_post(); ?>
            <h2><?php the_title() ?></h2>
            <?php the_author_posts_link() ?>
            <div class="thecontent">
               <?php the_content(); ?>
            </div>
         <?php endwhile; else: ?>
            <p>Oops! There are no posts to display.</p>
         <?php endif; ?>
```

3. Add the following line of code immediately above the loop:

```
<?php query_posts($query_string.'monthnum=10'); ?>
```

4. Save the file.

That's all there is to it. If you visit your site, you will now see that the page displays only the posts published in any October of any year.

How it works...

The `monthnum` parameter allows you to specify a month number, and retrieve only the posts published in the given period. You can also get posts by day, year, hour, and even minutes and seconds. In the following recipe, we will look at how to combine multiple parameters to narrow in on specific posts.

Here are the date and time parameters which can be used within the `query_posts()` function:

- `hour`: Number from 0 to 24, displays posts made during this time
- `minute`: Number from 0 to 59
- `second`: Number from 0 to 59
- `day`: Number from 0 to 31, shows all posts made, for example on the 15th
- `monthnum`: Number from 1 to 12
- `year`: Year, show all posts made in for example, 2007

Displaying posts published today

If you publish multiple posts per day, you may very well wish to display your most current posts together, in order to highlight the new content. The WordPress loop makes this possible, with a little help from some easy PHP and the `query_posts()` function. In this recipe, we show you how to display only those posts published today.

Getting ready

All you need to execute this recipe is your favorite code editor and access to the WordPress files on your server. You will also need a theme template file, which we will use to hold our modified WordPress loop. To keep this recipe short and to the point, we use the loop we created in the first recipe in this chapter.

How to do it...

Let's create a situation where the loop produces only the posts that were published today. Follow these steps:

1. Open the file that contains the loop. We'll use the same file we created in the first recipe in this chapter.
2. Find the loop:

```
         <?php if ( have_posts() ) : while ( have_posts() ) :
the_post(); ?>
            <h2><?php the_title() ?></h2>
            <?php the_author_posts_link() ?>
            <div class="thecontent">
               <?php the_content(); ?>
            </div>
         <?php endwhile; else: ?>
            <p>Oops! There are no posts to display.</p>
         <?php endif; ?>
```

3. Add the following lines of code immediately above the loop:

```
<?php
$current_day = date('j');
$current_month = date('m');
$year = date('Y');
query_posts($query_string . 'day='.$current_
day.'&month='.$current_month.'&year='.$year); ?>
```

4. Save the file.

If you visit your site, you will now see that the page displays only those posts published on the current date.

How it works...

In the preceding code, we started with creating PHP variables named `$current_day` `$current_month` and `$year`. We then used the `query_posts()` function to tell the system to only extract those posts that matched the date criteria.

Displaying posts published exactly one year ago

Here's a nice idea to give a second life to your older posts. This recipe automatically displays the posts you published exactly one year ago. The recipe uses some simple PHP, the WordPress loop, and the super-useful `query_posts()` function.

Getting ready

All you need to execute this recipe is your favorite code editor and access to the WordPress files on your server. You will also need a theme template file, which we will use to hold our modified WordPress loop. To keep this recipe short and to the point, we use the loop we created in the first recipe in this chapter.

How to do it...

Let's create a situation where the loop produces only the posts that were published exactly one year ago. Follow these steps:

1. Open the file that contains the loop. We'll use the same file we created in the first recipe in this chapter.

2. Find the loop:

```
         <?php if ( have_posts() ) : while ( have_posts() ) :
the_post(); ?>
            <h2><?php the_title() ?></h2>
            <?php the_author_posts_link() ?>
            <div class="thecontent">
               <?php the_content(); ?>
            </div>
         <?php endwhile; else: ?>
            <p>Oops! There are no posts to display.</p>
         <?php endif; ?>
```

3. Add the following lines of code immediately above the loop:

```
<?php
$current_day = date('j');
$current_month = date('m');
$last_year = date('Y')-1;
query_posts($query_string . 'day='.$current_
day.'&month='.$current_month.'&year='.$last_year); ?>
```

4. Save the file.

If you visit your site, you will now see that the page displays only those posts published on the same day of the previous year.

How it works...

The code in this recipe works exactly the same as that in the previous example. First, we have to use the PHP `date()` function to get the current day and month number, and then we subtract 1 from the current year. Once done, we simply use `query_posts()` and the WordPress loop to display the posts.

Using multiple loops

While the default implementation of the WordPress loop uses just one instance of the loop in a template, there is no reason why you cannot use multiple loops in a single template. Using multiple loops allows you the ability to produce specific content items, or lists of contents items in multiple locations on the page – in a sidebar, in the header, in the footer, and so on.

In this recipe, we explore the `WP_query()` function, and learn how to use it to create multiple loops inside a single template.

> Creating multiple loops does have an incremental impact on site performance, so while we say there is "no reason why you cannot use multiple loops," that statement comes with the caveat that you are not running an extreme number of loops on each page!

Getting ready

To execute this recipe you will need a code editor and access to the WordPress files on your server. You will also need a theme template file, which we will use to hold loops we create.

How to do it...

We are going to put two loops in a single template, so there are two parts to this recipe. We're going to set up one loop to show all the posts with the category ID 3. Then we will set up another loop to display the two most recent posts.

In the first part, we create the first loop, using `WP_query ()`. Follow these steps:

1. Open for editing the template file where you want the loops to appear.
2. Insert the following lines of code:

```
<?php // Loop 1
$first_query = new WP_Query('cat=9');
while($first_query->have_posts()) : $first_query->the_post(); ?>
<h1><a href="<?php the_permalink(); ?>"><?php the_title();
     ?></a></h1>
<?php the_content(); ?>
<?php endwhile; ?>
<?php wp_reset_postdata(); ?>
```

Next, let's set up the second loop. Follow these steps:

1. Add the following to the template file:

```
<?php // Loop 2
$second_query = new WP_Query('showposts=2');
while($second_query->have_posts()) : $second_query->the_post(); ?>
   <h1><a href="<?php the_permalink(); ?>"><?php the_title();
      ?></a></h1>
   <?php the_content(); ?>
<?php endwhile; ?>
<?php wp_reset_postdata(); ?>
```

2. Save the file.

If you view the page on your site now, you will see output from both of the loops.

How it works...

`WP_Query()` provides an alternative method for grabbing posts for display by the loop. It can be used to create any number of loops and is the preferred method for putting multiple loops in a single template. It accepts the same parameters as `query_posts()`, making it easy to use.

In the preceding code, we created first a single loop that uses the `cat` variable to pull out the posts in a specific category. The `while` statement cycles through the posts, producing them for output until there are no more posts that meet the criteria.

The second loop uses the `showposts` variable to limit the number of posts displayed to the two most recent and then again we see the `while` statement being used to output the posts.

Both loops close in standard fashion.

> Visit the WordPress Codex to learn more about `WP_Query()` at `http://codex.wordpress.org/Class_Reference/WP_Query`

Accessing post data outside of the WordPress loop

WordPress template tags, such as `the_title()` or `the_content()`, can't be used outside the loop. There will, however, be times that you want to be able to access post data from outside the loop, for example, to tap into author or post data outside of the primary content display of your page. To access post data outside the loop, you need to tap into the power of WordPress' global variables.

In this recipe, we'll show you how to access post data anywhere on your theme, without using the loop.

Getting ready

All you need to execute this recipe is your favorite code editor and access to the WordPress files on your server. You can use any template files from your active theme for this example.

How to do it...

1. Open the template file for editing.
2. Add the following code to the template file:

```
<?php setup_postdata($post); ?>
<?php the_title(); ?>
<?php the_content(); ?>
```

3. Save the file.

If you now visit the page on your site that uses the template, you should see the title and the body content of the first post in your system.

How it works...

The `$post()` function allows us to access the post data. Once called, we can use template tags to display the post data on the page, as we would inside the loop.

There's more...

WordPress is a very flexible system and there are alternative methods for accessing your post data outside the loop. While the `$post()` approach outlined in the recipe above is probably the simplest, there are other options.

Using get_post()

You can also access post data directly from the database by using the function `get_post()`. `get_post()` works entirely outside the loop. Nonetheless, as it is capable of querying the database, you can use it to grab multiple posts by a variety of criteria.

To display post data, you have to specify the name of the data that you'd like to show.

> Learn more about the function `get_post()` at `http://codex.wordpress.org/Function_Reference/get_post` and also here: `http://codex.wordpress.org/Template_Tags/get_posts`

The following post data is the available from the database:

- `post_author`: ID of the post author
- `post_date`: Publication date; according to your date settings
- `post_date_gmt`: Publication date; according to GMT time
- `post_content`: Post content

- `post_title`: Post title
- `post_category`: ID of the post category
- `post_excerpt`: Post excerpt
- `post_status`: Post status (published, draft, and so on)
- `comment_status`: If comments are open
- `ping_status`: If pingbacks are allowed
- `post_password`: Post password, if any
- `post_name`: Post permalink `%postname%`
- `to_ping`: Sites to ping
- `pinged`: Sites pinged
- `post_modified`: Last modified date; according to your time settings
- `post_modified_gmt`: Last modified date; according to GMT time
- `post_content_filtered`: Post content, filtered
- `post_parent`: ID of the post parent, if any
- `guid`: Standard URL (for example, `http://blog.com/?p=10`)
- `menu_order`: Order in the menu
- `post_type`: Post or page
- `post_mime_type`: Mime time of the post
- `comment_count`: Number of comments or trackbacks of the posts

The previous data should be used in the following way:

```
echo $data->post_title;
echo $data->post_date_gmt;
echo $data->post_content;
echo $data->post_category;
echo $data->post_excerpt;
echo $data->post_status;
echo $data->comment_status;
echo $data->ping_status;
echo $data->post_password;
echo $data->post_name;
echo $data->to_ping;
echo $data->pinged;
echo $data->post_modified;
echo $data->post_modified_gmt;
echo $data->post_content_filtered;
echo $data->post_parent;
echo $data->guid;
echo $data->menu_order;
echo $data->post_type;
echo $data->post_mime_type;
echo $data->comment_count;
```

Using get_post_data()

The function `get_post_data()` lets you access the data for a specific post. This approach differs, as you can see in the following steps:

1. Open your active theme's `functions.php` file for editing.
2. Add the following code to the file:

   ```
   function get_post_data($postId) {
      global $wpdb;
      return $wpdb->get_results("SELECT * FROM $wpdb->posts WHERE
   ID=$postId");
   }
   ```

3. Save the file.

> The global variable `$wpdb` is used to communciate with the WordPress database.

The function is now ready to use. To use the function, add the following anywhere in your theme files:

```
<?php
$data = get_post_data(5): // the ID number of the post you wish to
display all or part of
echo $data[0]->post_title; // Print the title
echo $data[0]->post_content; // Print the content
?>
```

Note that the function takes a single argument: the post ID. Use the ID of the post you wish to access to make all the elements of that post available to you. Once called, you can display all or part of the post data. While in the example above we only display the title and the content, you can display all the other elements of the post in a similar fashion.

> Use this technique sparingly as it does increase load on your database!

Accessing permalinks outside the loop

The previous recipe gave you access to post data outside the loop, but what if you need the permalink? Permalink data is dealt with differently and so you need a different approach.

In this recipe we show you how to use the `get_permalink()` function to get the permalink information from outside the WordPress loop.

Getting ready

All you need to execute this recipe is your favorite code editor and access to the WordPress files on your server. You can use any template files from your active theme for this example.

How to do it...

1. Get the post ID of the post for which you wish to display a permalink.
2. Open for editing the template file where you wish to display the permalink.
3. Add the following line of code:
   ```
   <a href="<?php echo get_permalink(10); ?>" >Link to the post</a>
   ```
4. Save the file.

That's all there is to it!

How it works...

Just like `get_post()`, the `get_permalink()` function takes a single argument—the ID of the post.

There's more...

When you're on a post or page, and within the loop, a `$post` global variable is initialized. This variable contains all of the data you can get with the `get_post()` function that we saw in the previous recipe. The `get_permalink()` function can be used along with the `$post` global variable, for example:

```
<a href="<?php echo get_permalink($post->ID); ?>" >Link
     to the post</a>
```

In order to minimize database requests, if you are inside the loop, you should always try to use the `$post` variable.

Displaying thumbnails on your homepage

You must have heard many times that 'A picture is worth a thousand words'. This old saying remains true on the web today. Many sites use thumbnails displayed next to post excerpts on their homepage to visually enhance their site and give it a more professional look.

In this recipe, we'll show you how you can easily add thumbnails to your homepage, independent of the theme you're using.

To add thumbnails to your homepage, we'll be using custom fields. Custom fields are one of the most powerful WordPress techniques. Using custom fields, you can define a key and give it a value. In your template, you simply have to get the key to display the custom value. Custom fields are individually defined on each post or page.

Getting ready

All you need to execute this recipe is your favorite code editor and access to the WordPress files on your server. You can use any template files from your active theme for this example.

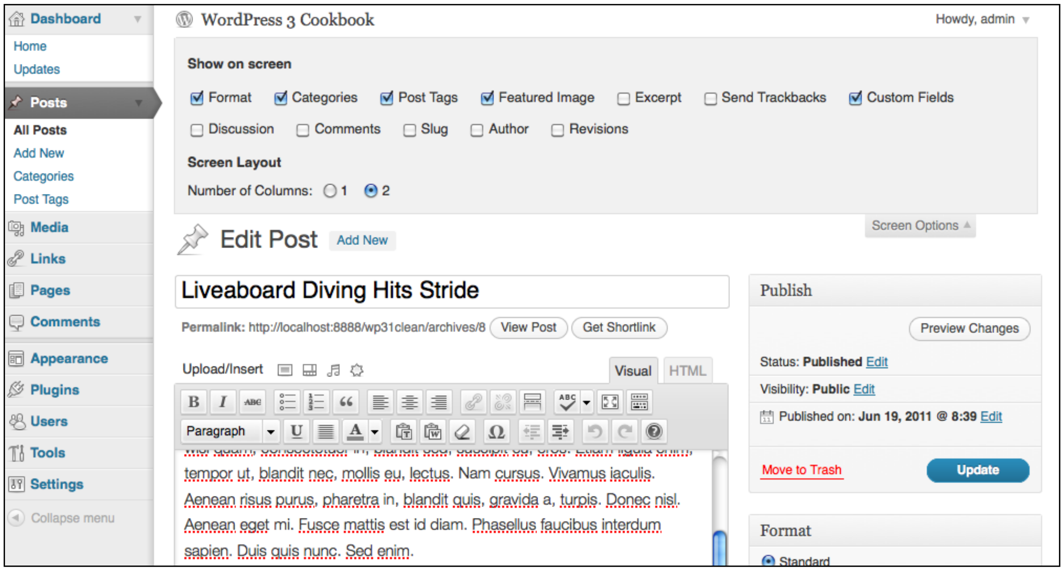
How to do it...

There are three parts to executing this recipe. First, we have to enable the custom fields option in the WordPress editor. Then we must create a key, and finally we have to get the key into the template files.

As of WordPress 3.1, some of the functionality in the editing window is hidden by default. One of the items that has been hidden is the custom fields functionality, which we need to complete this recipe. To enable the custom fields functionality, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Open a post for editing.
3. Click on the **Screen Options** tab at the top of the page.

4. Select the option **Custom Fields** option, as shown in the following screenshot:



Show on screen

☑ Format ☑ Categories ☑ Post Tags ☑ Featured Image ☐ Excerpt ☐ Send Trackbacks ☑ Custom Fields

☐ Discussion ☐ Comments ☐ Slug ☐ Author ☐ Revisions

Screen Layout

Number of Columns: ○ 1 ● 2

Liveaboard Diving Hits Stride

Permalink: http://localhost:8888/wp31clean/archives/8

tempor ut, blandit nec, mollis eu, lectus. Nam cursus. Vivamus iaculis. Aenean risus purus, pharetra in, blandit quis, gravida a, turpis. Donec nisl. Aenean eget mi. Fusce mattis est id diam. Phasellus faucibus interdum sapien. Duis quis nunc. Sed enim.

Status: **Published** Edit

Visibility: **Public** Edit

Published on: **Jun 19, 2011 @ 8:39** Edit

Dashboard
Home
Updates
Posts
All Posts
Add New
Categories
Post Tags
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Screen Options
Edit Post
Add New
View Post
Get Shortlink
Upload/Insert
Visual
HTML
Paragraph
Publish
Preview Changes
Move to Trash
Update
Format
Standard

5. Click on the **Screen Options** tab again to hide the panel from sight.

Now that the custom fields function is activated, we can move forward and set up the key that we need:

1. Open for editing the post where you wish to display the thumbnail.
2. Upload the thumbnail image for this post. Though it depends on your design, a good size is typically around 120 x 120 pixels.
3. Copy the image's URL.
4. Close the image upload dialogue.
5. Scroll down the page until you see **Custom fields**. In the **name** field enter **thumbnail**.
6. In the **value** field, paste the image URL that you previously copied.
7. Click on the **Add Custom Field** button.
8. **Update** the post.

If you visit your blog now, you won't see any changes. The custom field is defined and it has a value, but for now our theme doesn't know what to do with it. In the next steps, we tell the template to display the thumbnail:

1. Open your `index.php` file for editing.
2. Find the WordPress loop.

3. Add the following code into the loop, placing it where you want to see your thumbnails:

```
<img src="<?php echo get_post_meta($post->ID, "thumbnail",
true);?>" />
```

4. Save the file.

Visit your site and you should see the thumbnails displayed! Now, you can go back in and add styling around the code we inserted to control the formatting of the thumbnail, or even add other attributes to the `<img>` tag.

How it works...

To be fully functional, a custom field has to be entered on any post where you'd like to use it, and your theme must have the preceding code to handle it. Some recent WordPress themes can natively handle custom fields, but a majority of themes still can't.

To get the value of the custom field, I have used the `get_post_meta()` function, which takes the key name of your custom field as an argument. The function simply produces on the screen the value you gave to the key field for this particular post, in this case, as part of the URL string for an image.

Alternating background colors on post lists

A common technique used to improve the readability of long lists of posts is to employ what is commonly known as zebra striping, that is, to provide alternative background colors to the posts. The use of a subtle background color on every other post helps to visually separate the posts and makes it easier on the viewers' eyes.

In this recipe, we look at how to add alternating background colors to you post lists.

Getting ready

This recipe can be achieved with any WordPress theme. You don't need anything except a WordPress theme and a text editor.

How to do it...

1. Access the active theme files for your WordPress installation.
2. Open `index.php` for editing.
3. Find the loop.

4. Replace your current loop with the following one:

   ```
   <?php $odd_or_even = 'odd'; ?>
   <?php if (have_posts()) : while (have_posts()) : the_post(); ?>
   <div class="post <?php echo $odd_or_even; ?>">
   <?php $odd_or_even = ('odd'==$odd_or_even) ? 'even' : 'odd'; ?>
      <h2><?php the_title(); ?></h2>
      <?php the_content(); ?>
   </div>
   <?php endwhile; else: ?>
      <p>Sorry, no posts matched your      criteria.'</p>
   <?php endif; ?>
   ```

5. Save `index.php`.
6. Open your active theme's `style.css` file for editing.
7. Add the following classes to `style.css`:

   ```
      post.odd{
          background: #ccc;
      }
      .post.even{
          background: #fff;
      }
   ```

8. Save `style.css`

If you visit your site, you will now see your post backgrounds display alternating background colors. Feel free to replace the colors in the code with the colors of your choice.

How it works...

The principle used here is fairly simple. We alternate two CSS classes. To achieve this, we use a PHP operator, which is a short way to write:

```
<?php
if ('odd' == $odd_or_even){
    $odd_or_even = 'even';
}else{
    $odd_or_even = 'odd';
}
?>
```

Displaying posts in two columns

An increasing number of themes support two column layouts. If yours does not, this recipe will give you a solution. In this recipe, we add a bit of PHP logic to the loop and some CSS styling to the style sheet to achieve a ready-to-use two-column display of your posts.

Getting ready

All you need to execute this recipe is your favorite code editor and access to the WordPress files on your server. You can use any template files from your active theme for this example.

How to do it...

1. Access your WordPress installation.
2. Find the template file where you wish to use the two-column layout and open it for editing.
3. Find your current loop in the file.
4. Replace the loop with the following code:

```
<?php $col = 1; ?>
   <?php if (have_posts()) : while (have_posts()) :    the_post();
?>
   <?php if ($col == 1) echo "<div class=\"row\">"; ?>
      <div class="post hol<?php echo $col;?>" id="post-<?php
the_ID(); ?>">
            <h2><?php the_title(); ?></h2>
            <?php the_excerpt(); ?>
      <?php if ($col == 1) echo "</div>";(($col==1) ? $col=2 :
$col=1); ?>
      </div>

<?php endwhile; else: ?>
   <p>Sorry, no posts matched your   criteria.'</p>
<?php endif; ?>
```

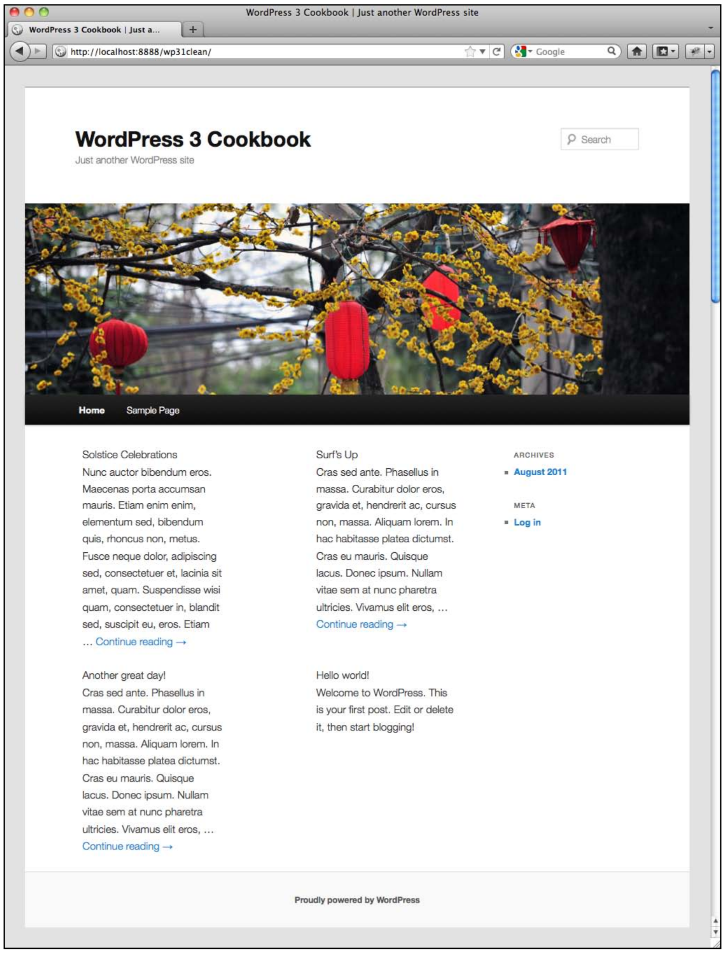
5. Save the file.
6. Open the `style.css` file of your active theme for editing.
7. Add the following styles to the file:

```
.row {
   clear: both;
}
```

```
.hol1 {
   width: 200px;
   float: left; padding: 0 10px;
}

.hol2 {
   width: 200px;
   float: right;
   padding: 0 10px;
}
```

8. Save your `style.css` file.

If you visit your site, you will now see that your posts are displayed in two columns as seen in the following screenshot:


WordPress 3 Cookbook | Just another WordPress site
WordPress 3 Cookbook | Just a...
http://localhost:8888/wp31clean/
Google
Search
Home
Sample Page
ARCHIVES
August 2011
META
Log in

WordPress 3 Cookbook

Just another WordPress site

Solstice Celebrations

Nunc auctor bibendum eros. Maecenas porta accumsan mauris. Etiam enim enim, elementum sed, bibendum quis, rhoncus non, metus. Fusce neque dolor, adipiscing sed, consectetuer et, lacinia sit amet, quam. Suspendisse wisi quam, consectetuer in, blandit sed, suscipit eu, eros. Etiam … Continue reading →

Surf's Up

Cras sed ante. Phasellus in massa. Curabitur dolor eros, gravida et, hendrerit ac, cursus non, massa. Aliquam lorem. In hac habitasse platea dictumst. Cras eu mauris. Quisque lacus. Donec ipsum. Nullam vitae sem at nunc pharetra ultricies. Vivamus elit eros, … Continue reading →

Another great day!

Cras sed ante. Phasellus in massa. Curabitur dolor eros, gravida et, hendrerit ac, cursus non, massa. Aliquam lorem. In hac habitasse platea dictumst. Cras eu mauris. Quisque lacus. Donec ipsum. Nullam vitae sem at nunc pharetra ultricies. Vivamus elit eros, … Continue reading →

Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!

Proudly powered by WordPress

How it works...

Before the loop starts, we initialized a PHP variable called `$col`. This variable will contain a value (1 or 2) that helps differentiate the posts and decide when to start a new row or continue with the existing one. We use divs to style the posts into a two-column layout.

After the loop displays the post data, the `$col` variable checks whether to close the `div` tag; the value of the variable determines which column the post goes into. Finally, we used a ternary operator to give `$col` manage the value of variable.

Save time by using WordPress shortcodes

When you know that you'll have to insert the same code snippets on many posts, you should definitely create a shortcode. Introduced in WordPress 2.5, the shortcode API provides a simple set of functions for creating macro codes for use in post content. A classic shortcode looks like this:

```
[author_info]
```

Shortcodes can handle attributes. For example:

```
[download file="myfile.zip"]
```

Also, a shortcode can have embedded content:

```
[mycode]Some Content[/mycode]
```

In this recipe, we show you how you can create your own shortcodes and then use them in posts.

Getting ready

This recipe can be achieved with any WordPress theme. You don't need anything except your favorite WordPress theme and a text editor.

How to do it...

Creating shortcodes is relatively simple. For your first shortcode, let's create a simple one that will display a disclaimer.

We begin by adding a simp9le PHP function to your theme's `functions.php` file. To do so, follow these steps:

1. Open for editing the `functions.php` file from your active theme.

2. Add the following code to the file:

```
function displayDisclaimer() {
   return 'This product is meant for educational purposes only.
      Use of the programs or procedures in such a manner, is at
       your own risk.';
}
add_shortcode('disclaimer', 'displayDisclaimer');
```

You're now able to use the `disclaimer` shortcode. To employ the shortcode, follow these steps:

1. Write a new post, or edit an existing one.
2. Click on the **HTML** tab to switch the editor to HTML mode.
3. Insert the shortcode you just created:

   ```
   [disclaimer]
   ```

4. Click on the **Update** button.

If you visit the page you just edited, instead of seeing `[disclaimer]`, you will see the text you put into the function.

How it works...

In this example, we created a very basic function and then used the `add_shortcode()` function to turn it into a shortcode. The `add_shortcodes()` function takes two arguments—the first is the shortcode name, and the second is the function to call when the shortcode will be used.

Every time WordPress displays a post, the system automatically looks for shortcodes and, when it finds one, executes the associated function.

Enabling the use of shortcodes in widgets

While WordPress shortcodes are incredibly useful, sadly, they are only available inside of posts and pages. In this recipe, we show you how to extend the power of shortcodes by enabling their use in widgets.

Getting ready

This recipe requires modifying the `functions.php` file of your theme. This will work with any theme, so all you need is access to your theme files and a code editor.

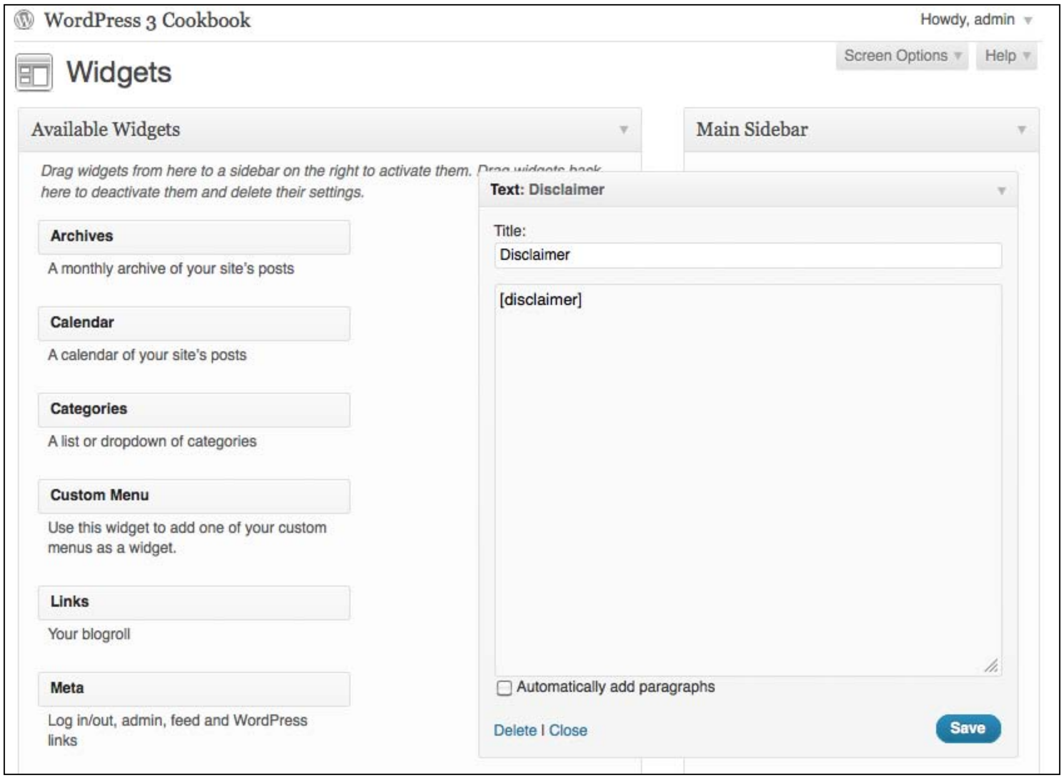
How to do it...

1. Access your active WordPress theme.
2. Open the `functions.php` file for editing.
3. Add the following line of code to the file:

```
add_filter('widget_text', 'do_shortcode');
```

4. Save the file.

That's all it takes. Now you can add shortcodes directly into a Text widget, as shown in the next screenshot, and the system will display the proper output:



WordPress 3 Cookbook
Howdy, admin
Screen Options
Help
Widgets
Available Widgets
Main Sidebar
Text: Disclaimer
Archives
Calendar
Categories
Custom Menu
Links
Meta
Delete | Close
Save

Drag widgets from here to a sidebar on the right to activate them. Drag widgets back here to deactivate them and delete their settings.

A monthly archive of your site's posts

A calendar of your site's posts

A list or dropdown of categories

Use this widget to add one of your custom menus as a widget.

Your blogroll

Log in/out, admin, feed and WordPress links

Title: Disclaimer

[disclaimer]

☐ Automatically add paragraphs

How it works...

The line of code added to the `functions.php` file simply allows the shortcodes to be properly interpreted by the system when they appear inside of widget content areas. Thereafter, normal shortcode usage applies and the system will display the proper output, rather than the shortcode text.

Adding notes to your posts

If you are working with others to create content for your site, there are times when it would be nice to be able to leave private notes concerning the revision of the content items. In this recipe, we look at using a shortcode to enable administrators to post private messages on content items. The messages can only be seen by other users who have the permission to publish posts, and even then, only when they are logged in. Public visitors to the site and registered users who do not have permissions to publish content will not see the messages. This gives you an easy way make comments to the author and provide suggestions and feedback directly on the content item.

Getting ready

All you need to execute this recipe is a code editor and access to your theme files. We will be working with the `nctions.php` file from your active theme.

How to do it...

First, let's create the shortcode:

1. Access the active theme files on your WordPress installation.
2. Open for editing the `functions.php` file.
3. Add to the file the following lines of code:

```
function admin_note( $atts, $content = null ) {
   if ( current_user_can( 'publish_posts' ) )
      return '<div class="adminnote">'.$content.'</div>';
   return '';
}
```

4. Save the file.

The shortcode is now ready to use. Next, let's create a unique style for the note so that it is easily visible. Follow these steps:

Open for editing the `style.css` file of your active theme.

1. Add the following selector to the file:

```
.adminnote {
   color:#990000;
   background:#ffff00;
}
```

2. Save the `style.css` file.

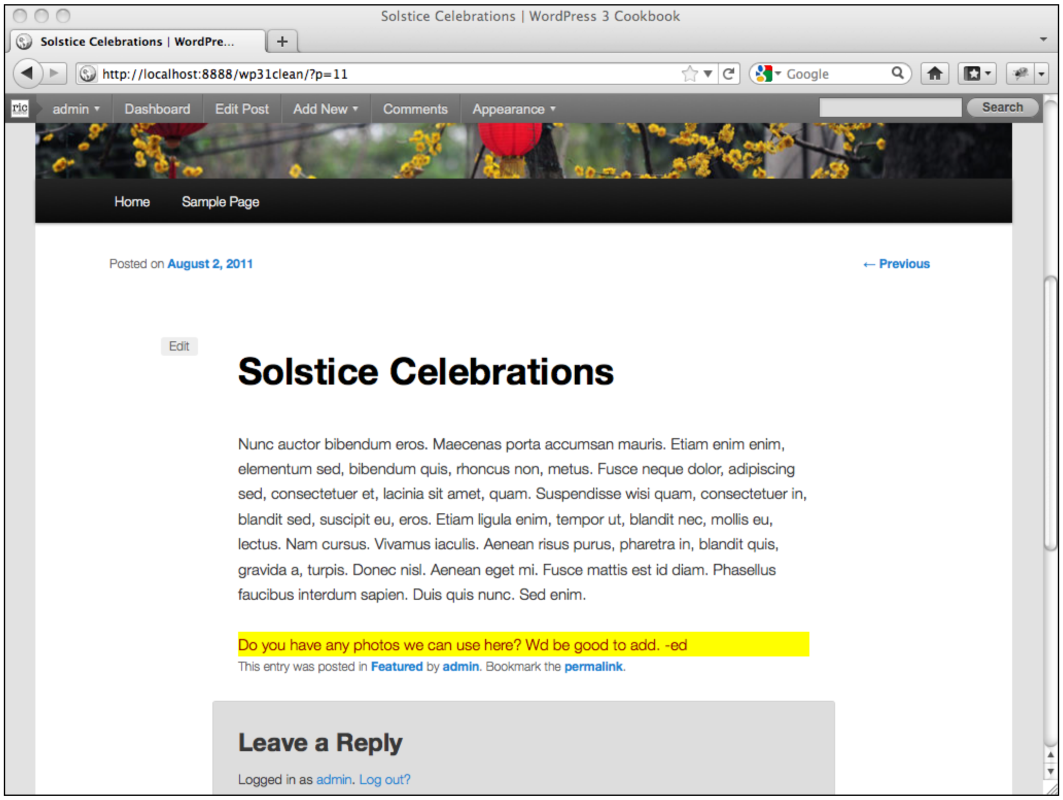
To apply the new admin note shortcode to a post, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Open a content item for editing, or create a new item.
3. Click on the **HTML** tab to switch to the HTML editor.
4. Type your message into the editing window, using a format like this:

   [adminnote] Your message goes in here [/adminnote]

5. Click on the **Update** button to save the content item.

If you stay logged in and visit the page on your site, you will now see the message, as shown in the following screenshot:


Solstice Celebrations | WordPress 3 Cookbook
Solstice Celebrations | WordPre...
http://localhost:8888/wp31clean/?p=11
Google
admin
Dashboard
Edit Post
Add New
Comments
Appearance
Search
Home
Sample Page
Posted on August 2, 2011
← Previous
Edit

Solstice Celebrations

Nunc auctor bibendum eros. Maecenas porta accumsan mauris. Etiam enim enim, elementum sed, bibendum quis, rhoncus non, metus. Fusce neque dolor, adipiscing sed, consectetuer et, lacinia sit amet, quam. Suspendisse wisi quam, consectetuer in, blandit sed, suscipit eu, eros. Etiam ligula enim, tempor ut, blandit nec, mollis eu, lectus. Nam cursus. Vivamus iaculis. Aenean risus purus, pharetra in, blandit quis, gravida a, turpis. Donec nisl. Aenean eget mi. Fusce mattis est id diam. Phasellus faucibus interdum sapien. Duis quis nunc. Sed enim.

Do you have any photos we can use here? Wd be good to add. -ed

This entry was posted in Featured by admin. Bookmark the permalink.

Leave a Reply

Logged in as admin. Log out?

How it works...

We started in the `functions.php` file, where we created a new function called `admin_note()`. In that function we tell the system to check whether the active user has the permission to publish posts and if so, we show that user any content inside the adminnote shortcode. Next, we edited the `style.css` file to provide a selector for the class we used to wrap the `$content` in the `functions.php` file.

Adding tags to your pages

The tags system included in the WordPress core makes it easy for you to tag your posts and then produce a tag cloud, based on that data. Unfortunately, the default tagging system does not extend to the pages of your site – it only works on posts.

In this recipe we look at implementing a simple plugin to fill that gap and to bring full tagging to your site's pages.

Getting ready

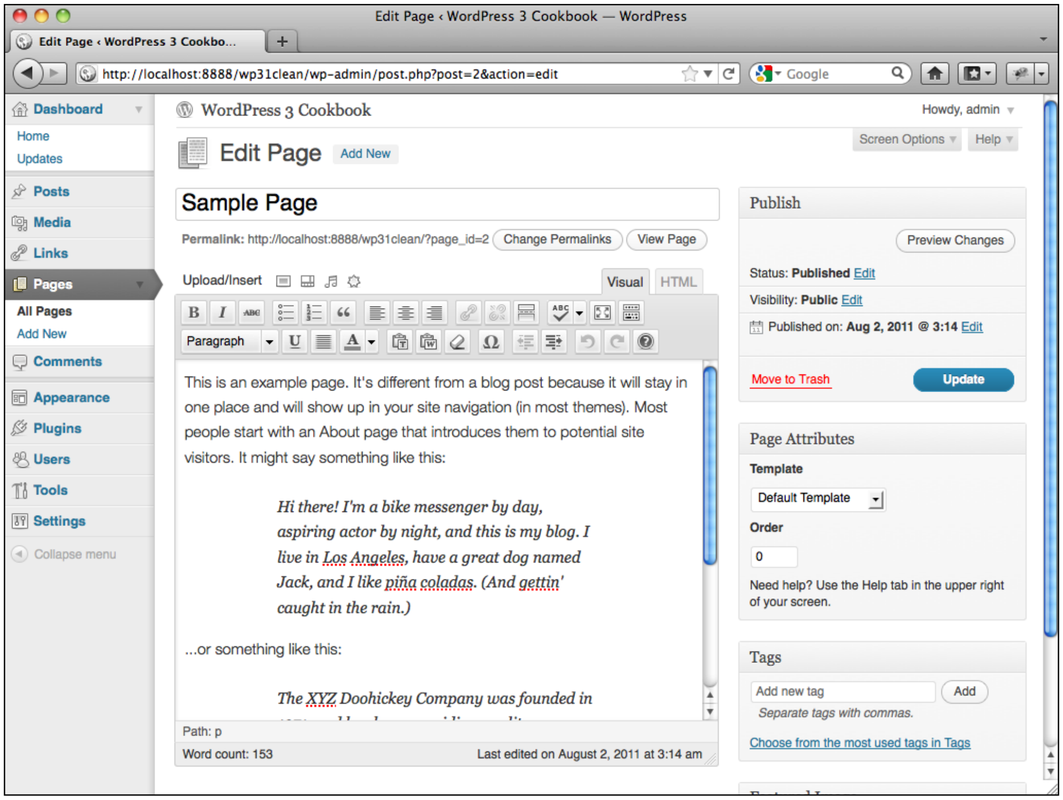
To execute this recipe, you will need to install the Page Tagger plugin. You will need to install this plugin before you can get started. Search for page tagger inside the **Add New Plugins** screen of your WordPress site. After you find it, click to install, and then activate it.

> You can learn more about the plugin by visiting the developer's site at `http://www.hiddentao.com/code/wordpress-page-tagger-plugin/`

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Pages** menu.
3. Click on the page your wish to edit, or create a new page.
4. In the right hand column, typically below the **Page Attributes** widget, you will see the **Tags** widget. The next screenshot shows the **Tags** widget. Simply enter the tags you wish to associate with the page, separating multiple tags with commas.
5. Click on the **Add** button to save the tags.

6. Click on the **Update** button to save the page.


Edit Page ‹ WordPress 3 Cookbook — WordPress
Edit Page ‹ WordPress 3 Cookbo...
http://localhost:8888/wp31clean/wp-admin/post.php?post=2&action=edit
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
All Pages
Add New
Comments
Appearance
Plugins
Users
Tools
Settings
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Screen Options
Help
Edit Page
Add New
Change Permalinks
View Page
Upload/Insert
Visual
HTML
Paragraph
Publish
Preview Changes
Move to Trash
Update
Page Attributes
Template
Order
Tags
Add new tag
Add
Separate tags with commas.
Choose from the most used tags in Tags

Sample Page

Permalink: http://localhost:8888/wp31clean/?page_id=2

This is an example page. It's different from a blog post because it will stay in one place and will show up in your site navigation (in most themes). Most people start with an About page that introduces them to potential site visitors. It might say something like this:

> *Hi there! I'm a bike messenger by day, aspiring actor by night, and this is my blog. I live in Los Angeles, have a great dog named Jack, and I like piña coladas. (And gettin' caught in the rain.)*

...or something like this:

> *The XYZ Doohickey Company was founded in*

Path: p

Word count: 153 Last edited on August 2, 2011 at 3:14 am

Status: **Published** Edit

Visibility: **Public** Edit

Published on: **Aug 2, 2011 @ 3:14** Edit

Template: Default Template

Order: 0

Need help? Use the Help tab in the upper right of your screen.

How it works...

The plugin here does all the work; it merely extends the tagging functionality to the pages of your site.

5

Building Interactivity and Community

In this chapter, we will cover:

- Improving navigation with a paginator
- Highlighting searched text in search results
- Integrating a forum into your site
- Adding social bookmarking buttons to your site
- Aggregating RSS content
- Integrating Feedburner into your site
- Displaying a retweet button on your posts
- Getting more comments with the Subscribe to Comments Reloaded plugin
- Removing the nofollow attribute to motivate users to leave comments
- Providing recognition for your top contributors
- Displaying author-related information on posts
- Displaying the author's avatar on posts
- Allowing multiple authors on posts
- Diaplying a list of all of the authors
- Creating community with BuddyPress
- Adding a simple gallery to your site
- Bringing Facebook functionality into your site
- Integrating a Twitter stream into your site

Introduction

Websites today are expected to provide users with a degree of interactivity. Moreover, with the growth of social networks and the prevalence of mobile devices and location services, the ability for a site to generate community is becoming an increasingly important factor. In this chapter, we look at a variety of techniques and plugins that allow you to add important interactivity to your WordPress and enable you to build community.

Some of the recipes in this chapter are simple to execute, and others, such as the BuddyPress recipe, are quite complex. Before you consider the suggestions given in the following recipes, you should think carefully about what features your site visitors are likely to use.

> If you are unsure about how to begin with building community on your site, start small and add features as you learn what your users like and find beneficial.

The recipes in this chapter also include plugins that let you tap into two of today's most popular third-party platforms: Facebook and Twitter. There is some overlap in a few of these recipes, at least where they concern social bookmarking. The recipes that provide multiple social bookmarking solutions tend to be more general in functionality, and the recipes focused on specific systems tend to offer a richer feature set – choose whichever you feel is most suitable for you.

Improving navigation with a paginator

When a website, or blog, publishes lots of articles on a single page, the list can quickly become very long and hard to read. To solve this problem, paginations were created. Simple paginations allow you to move forward and backwards through the list of articles by clicking on next and previous links. More advanced pagination options present you with a numbered list of pages corresponding to the total number of articles (divided by the number of articles per page). Numbered pagination not only provides the user with more information about the number of pages and where they are in the list, it also makes it possible to jump directly to specific pages.

The default WordPress system relies on simple pagination. In this recipe, we'll show you how to integrate a plugin that enables the richer, numbered pagination option.

Getting ready

To execute this recipe, you will need to install the WP-Paginate plugin. You will need to install this plugin before you can get started. Search for **WP-Paginate** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

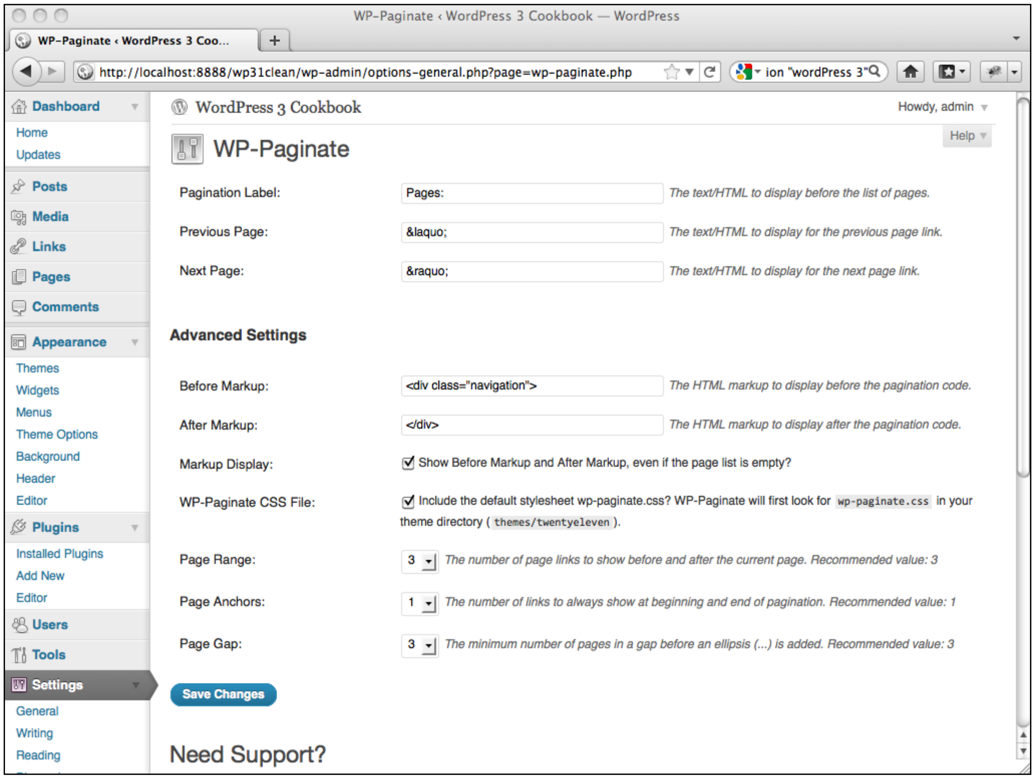
> You can learn more about the plugin by visiting the developer's site at `http://www.ericmmartin.com/projects/wp-paginate/`

How to do it...

There are two parts to the recipe. In the first part, we need to configure the plugin. In the second part, we need to add a bit of code to our theme to place the output on the page.

To begin, let's look at the configuration options offered by the plugin:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the **WP-Paginate** option.
4. On the page that appears, select the options you want to use. The image following shows the standard configuration:


WP-Paginate ‹ WordPress 3 Cookbook — WordPress
WP-Paginate ‹ WordPress 3 Coo...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=wp-paginate.php
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Theme Options
Background
Header
Editor
Plugins
Installed Plugins
Add New
Editor
Users
Tools
Settings
General
Writing
Reading
WordPress 3 Cookbook
Howdy, admin
Help
Save Changes
Need Support?

WP-Paginate

Pagination Label: Pages: — The text/HTML to display before the list of pages.

Previous Page: &laquo; — The text/HTML to display for the previous page link.

Next Page: &raquo; — The text/HTML to display for the next page link.

Advanced Settings

Before Markup: <div class="navigation"> — The HTML markup to display before the pagination code.

After Markup: </div> — The HTML markup to display after the pagination code.

Markup Display: ☑ Show Before Markup and After Markup, even if the page list is empty?

WP-Paginate CSS File: ☑ Include the default stylesheet wp-paginate.css? WP-Paginate will first look for `wp-paginate.css` in your theme directory (`themes/twentyeleven`).

Page Range: 3 — The number of page links to show before and after the current page. Recommended value: 3

Page Anchors: 1 — The number of links to always show at beginning and end of pagination. Recommended value: 1

Page Gap: 3 — The minimum number of pages in a gap before an ellipsis (...) is added. Recommended value: 3

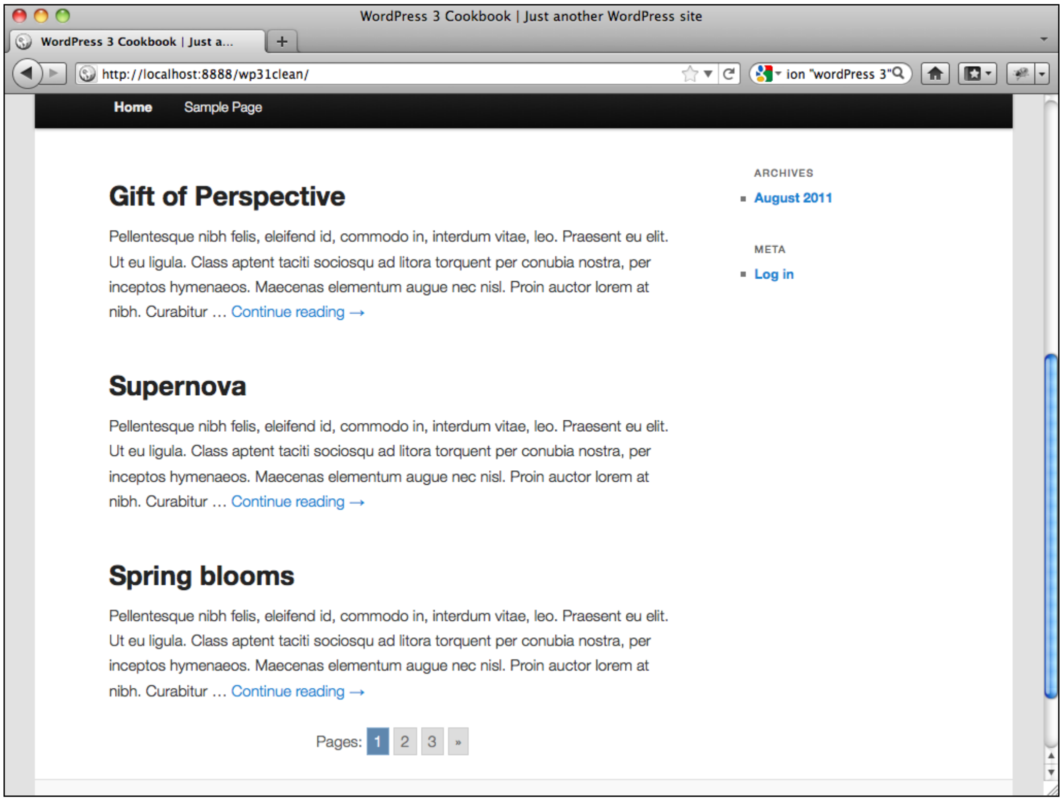
5. When you've finished, click on the **Save Changes** button.

The plugin is now ready to go; all that remains is for us to add the code to our theme files to display the pagination. Follow these steps:

1. Open the file containing the WordPress loop. For many this will be the `index.php` file. For some of the more recent themes, it may be the `loop.php` file.
2. Place the following code where you want the pagination controls to appear:

```
<?php if(function_exists('wp_paginate')) {     wp_
paginate(); } ?>
```

3. Save the file.
4. If you view your site, you should now see your new pagination controls, similar to what is shown in the following image:


WordPress 3 Cookbook | Just another WordPress site
WordPress 3 Cookbook | Just a...
http://localhost:8888/wp31clean/
ion "wordPress 3"
Home
Sample Page
ARCHIVES
August 2011
META
Log in

Gift of Perspective

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Maecenas elementum augue nec nisl. Proin auctor lorem at nibh. Curabitur … Continue reading →

Supernova

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Maecenas elementum augue nec nisl. Proin auctor lorem at nibh. Curabitur … Continue reading →

Spring blooms

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Maecenas elementum augue nec nisl. Proin auctor lorem at nibh. Curabitur … Continue reading →

Pages: 1 2 3 »

> If your theme relies on multiple template files, you may need to add the code to more than one file, for example, the templates that control the archive pages or the category pages.

How it works...

The plugin does most of the work here, but you still need to add the code to your template files so that the controls are displayed. The key for you is the placement of the code inside the template files in order to display the pagination controls where you want. You will also likely want to wrap the WP-Paginate code with a `div` or some other styling in order to control the appearance and exact placement of the controls.

Highlighting searched text in search results

The default WordPress search is somewhat limited. While it does a fine job of searching your content, it does not provide automatic highlighting of search terms in the search results; a useful feature that improves usability and makes it easier for users to identify the right article for their needs.

In this recipe, we look at installing a plugin that not only adds highlighting of search terms for searches made on your site, but also highlights search terms when visitors arrive at your site from the big search engines.

Getting ready

To execute this recipe, you will need to install the Highlight Search Terms plugin. You will need to install this plugin before you can get started. Search for "highlight search terms" inside the Add New Plugins screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's site at `http://4visions.nl/wordpress-plugins/highlight-search-terms/`

How to do it...

After you have installed and activated the plugin, you still need to take a further step. While the plugin is operational as soon as it is installed, the highlights will not appear on your site until you add a highlights style to your style sheet. Here's how to do it:

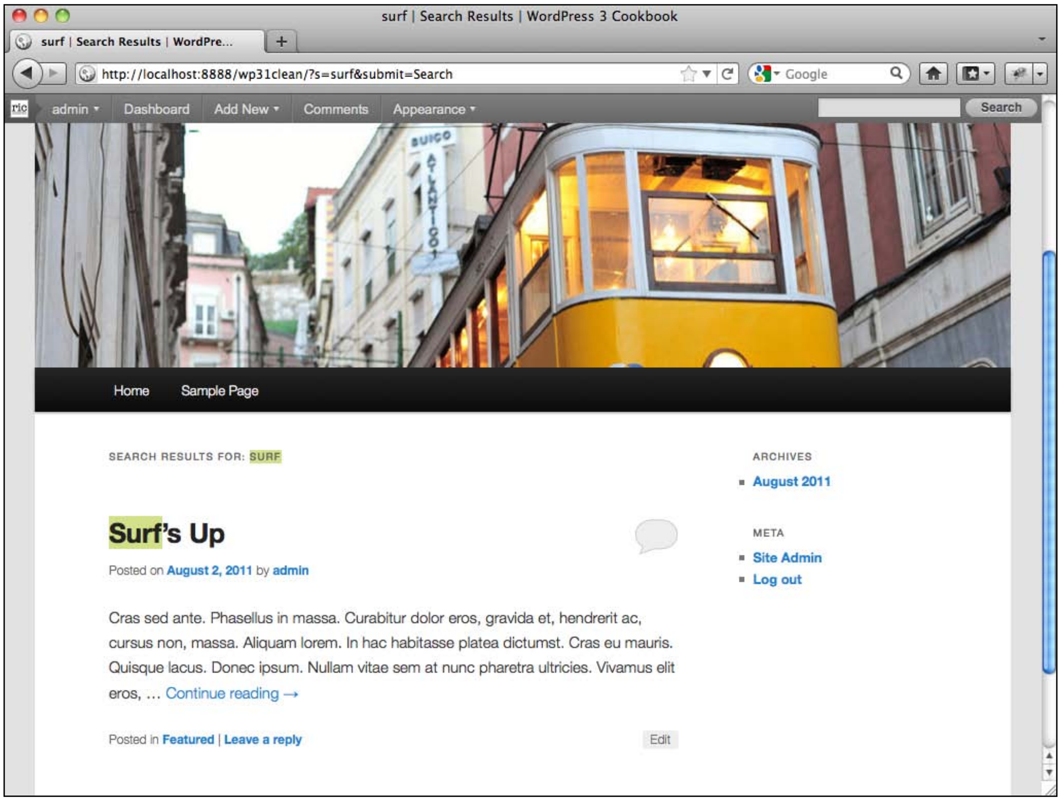
1. Access the theme files of your WordPress installation.
2. Open the `style.css` file of your active theme for editing.

3. Add the following selector to the stylesheet:

```
hilite {
   background:#D3E18A;
}
```

4. Save the file.

You're done. Now, the searched text will be highlighted in your search results, as shown in the following screenshot:


surf | Search Results | WordPress 3 Cookbook
surf | Search Results | WordPre...
http://localhost:8888/wp31clean/?s=surf&submit=Search
Google
admin
Dashboard
Add New
Comments
Appearance
Search
Home
Sample Page
ARCHIVES
August 2011
META
Site Admin
Log out
Edit

SEARCH RESULTS FOR: SURF

Surf's Up

Posted on August 2, 2011 by admin

Cras sed ante. Phasellus in massa. Curabitur dolor eros, gravida et, hendrerit ac, cursus non, massa. Aliquam lorem. In hac habitasse platea dictumst. Cras eu mauris. Quisque lacus. Donec ipsum. Nullam vitae sem at nunc pharetra ultricies. Vivamus elit eros, … Continue reading →

Posted in Featured | Leave a reply

How it works...

The plugin uses jQuery to power the highlights function. Highlights will show for the search results generated by your WordPress search and they will also show when someone runs a search on Google, Bing, Baidu, and other major search engines, then clicks on a link to your site in the search results; when the user arrives at your site, the terms they searched for will be highlighted in the content item.

The CSS selector we created uses a light green color (indicated by the hex value #D3E18A) for the background of the search terms. You can set the color value for any color that suits your theme.

Integrating a forum into your site

Adding a forum to your site is a great way to provide a place for users to interact with you and with each other. Forums are an effective means of stimulating interaction and increasing page views but they can, however, also increase site management overhead as forum administration can be a huge time sink. Moreover, if your site lacks sufficient traffic, forums can wither and die from lack of activity.

Many forum solutions are available for WordPress: from stand-alone systems like PhpBB/Vbulletin and BBPress to plugins that give you a native WordPress solution. One of the best plugin options for WordPress 3 is the WP Forum Server. The plugin is complete, powerful, and easy to install, use, and manage.

In this recipe, you're going to learn how to integrate WP Forum Server on your WordPress site.

Getting ready

To execute this recipe, you will need to install the WP Forum Server plugin. You will need to install this plugin before you can get started. Search for **WP Forum Server** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's site at `http://forumpress.org/`

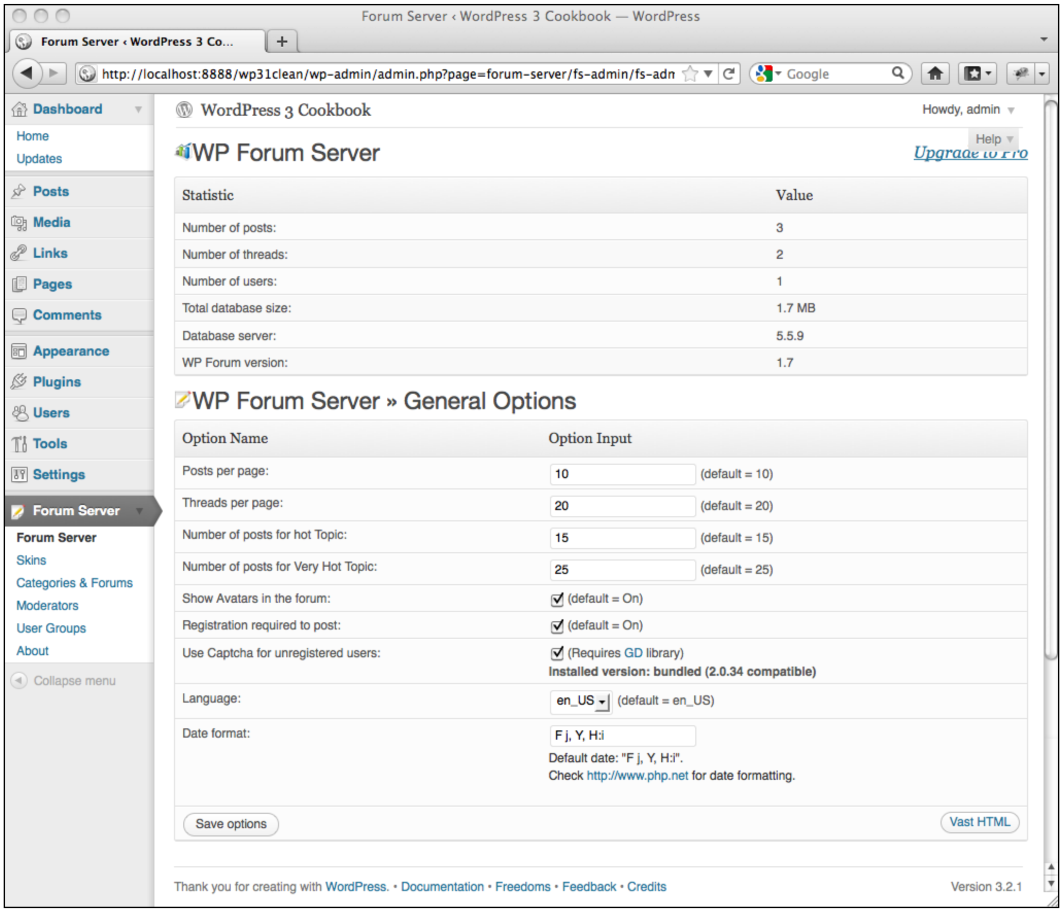
How to do it...

There are several parts to this recipe. In addition to configuring the plugin, you will also want to change the page template associated with the forum page.

Let's get started by configuring the plugin.

1. Log in to your WordPress **Dashboard**.
2. Click the menu named **Forum Server**.

3. On the page that appears, as shown in the following screenshot, you can set your basic forum options. Select the choices you want here.


Forum Server ‹ WordPress 3 Cookbook — WordPress
Forum Server ‹ WordPress 3 Co...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=forum-server/fs-admin/fs-adn
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Forum Server
Forum Server
Skins
Categories & Forums
Moderators
User Groups
About
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
Upgrade to Pro

WP Forum Server

Statistic	Value
Number of posts:	3
Number of threads:	2
Number of users:	1
Total database size:	1.7 MB
Database server:	5.5.9
WP Forum version:	1.7

WP Forum Server » General Options

Option Name	Option Input
Posts per page:	10 (default = 10)
Threads per page:	20 (default = 20)
Number of posts for hot Topic:	15 (default = 15)
Number of posts for Very Hot Topic:	25 (default = 25)
Show Avatars in the forum:	☑ (default = On)
Registration required to post:	☑ (default = On)
Use Captcha for unregistered users:	☑ (Requires GD library) Installed version: bundled (2.0.34 compatible)
Language:	en_US (default = en_US)
Date format:	F j, Y, H:i Default date: "F j, Y, H:i". Check http://www.php.net for date formatting.

Save options
Vast HTML
Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits
Version 3.2.1

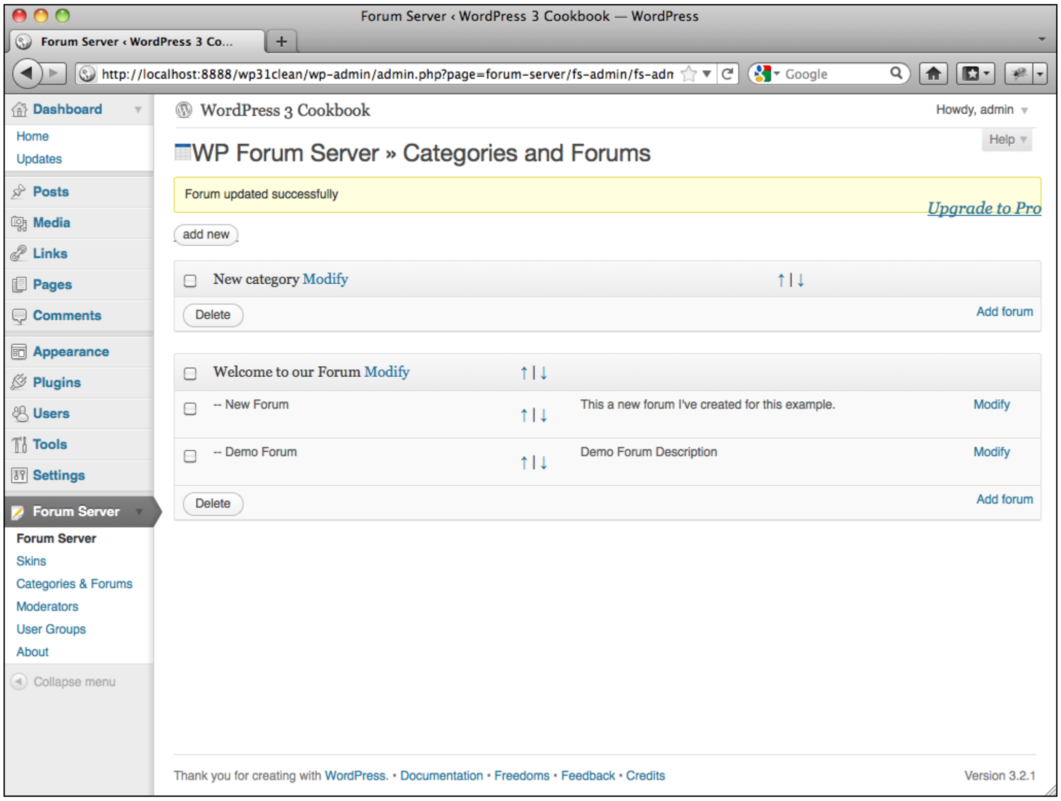
Next, let's set up some basic categories and forums to get things started.

1. On the **Forum Server** menu, click on the option **Categories & Forums**.

> In WP Forum Server terminology, categories are created to contain forums. Forums contain threaded discussion topics. If you only have one forum on your site, you only need one category.

2. You may want to begin by changing the name of the default **Category**, which is **uncategorized**. For our example, we changed it to a generic **Welcome to our Forum**. To make this change, click the **Modify** link next to the name of the category.

3. Type the label you desire in the field marked **Name**.
4. Add a description if you want.
5. Click on the **Save Group** button.
6. Click on the **Add forum** link to create new top-level forums into which your users can add topics.
7. Give the new forum a **Name** and a **Description**.
8. Click on the **Save forum** button. At the end of this process you will see something similar to what is shown in the following screenshot:



Forum Server ‹ WordPress 3 Cookbook — WordPress
Forum Server ‹ WordPress 3 Co...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=forum-server/fs-admin/fs-adn
Google
WordPress 3 Cookbook
Howdy, admin
Help
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Forum Server
Forum Server
Skins
Categories & Forums
Moderators
User Groups
About
Collapse menu
WP Forum Server » Categories and Forums
Upgrade to Pro
add new
Delete
Add forum
Modify
Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits
Version 3.2.1

Forum updated successfully

New category Modify

Welcome to our Forum Modify

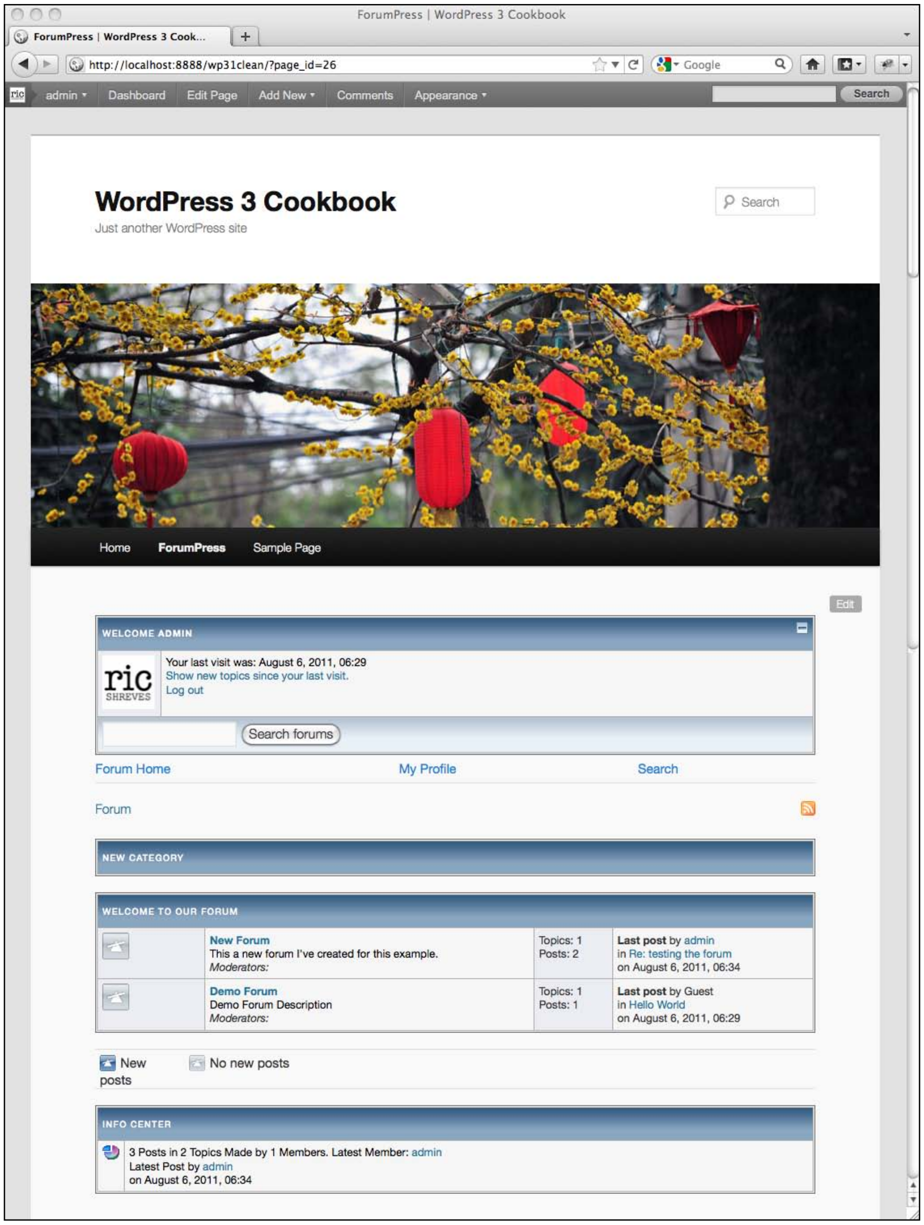
☐	-- New Forum	This a new forum I've created for this example.	Modify
☐	-- Demo Forum	Demo Forum Description	Modify

Your forum is now ready to go. When you installed this plugin, the system automatically created and published a new page containing the forum. In the default theme, the template that is automatically used by the system is not ideal – it is a two-column layout that forces the forum into a narrow column. Accordingly, we're going to change the template to allow the forum to span the width of the page. To do so, follow these steps:

1. Click the menu option **Pages**.
2. Click on the **Edit** link for the page named **ForumPress**.

3. Select from the **Template** combo box the option **Showcase Template**.
4. Click on the **Update** button.

If you access the front-end of your site and click on the menu option **ForumPress**, you will see your new forum in action, as shown in the following screenshot:


ForumPress | WordPress 3 Cookbook
ForumPress | WordPress 3 Cook...
http://localhost:8888/wp31clean/?page_id=26
Google
admin
Dashboard
Edit Page
Add New
Comments
Appearance
Search
WordPress 3 Cookbook
Just another WordPress site
Search
Home
ForumPress
Sample Page
Edit
WELCOME ADMIN
ric SHREVES
Search forums
Forum Home
My Profile
Search
Forum
NEW CATEGORY
WELCOME TO OUR FORUM
New posts
No new posts
INFO CENTER

Your last visit was: August 6, 2011, 06:29
Show new topics since your last visit.
Log out

	New Forum This a new forum I've created for this example. *Moderators:*	Topics: 1 Posts: 2	**Last post** by admin in Re: testing the forum on August 6, 2011, 06:34
	Demo Forum Demo Forum Description *Moderators:*	Topics: 1 Posts: 1	**Last post** by Guest in Hello World on August 6, 2011, 06:29

3 Posts in 2 Topics Made by 1 Members. Latest Member: admin
Latest Post by admin
on August 6, 2011, 06:34

How it works...

When installed, the plugin creates some new database tables in order to record threads and topics of your forum. For the rest, it is an advanced forum working the same way (but with less functionality) as popular solutions such as V-Bulletin or PhpBB.

There's more...

As you have seen in the **Forum Server** menu item, there are additional options that you can use to further customize this plugin. In this section we take a quick look at each.

Skins

Skins are CSS files and images that can give a new look and feel to your forum. There are multiple options built into the system, but you can also create new skins. To create a new skin, follow these steps:

1. Access your WordPress installation on the server.
2. Go to the skin's directory for the plugin, at `/wp-content/plugins/forum-server/skins`.
3. Create a new directory; name it for your new skin.
4. Copy the `style.css` file from an existing forum skin.
5. Paste it into your new skin's directory.
6. Open the `style.css` file for editing.
7. Change the name, description, and author information at the top of the file to reflect the name of the new skin and your details.
8. Edit the styles as you see fit to tailor the appearance of the new skin.
9. Save the file.

If you visit the **Skins** link on the **Forum Server** menu, you will see your new skin listed and ready to use.

Moderators

On this page, you can select one of your registered users (who must have the right to edit posts) and allow him to moderate your forum. You can create a global moderator (who can moderate all forums) or a normal moderator (who moderates one forum). Once you have chosen the user to be the moderator, just click on the **Add moderator** button to give him or her the moderation rights.

User groups

User groups allow you to group your users together for the purpose of giving the group specific permissions.You can create a group by naming it and giving it a description. Once done, you can add members to the group.

See also...

- *Creating community with BuddyPress*

Adding social bookmarking buttons to your theme

Social bookmarking websites are common features on today's web. Digg, Del.icio.us, StumbleUpon, Reddit, and many more. Such websites can drive a huge amount of traffic to your site. Given their potential value to your site, you should add social bookmarking buttons to your theme and give your readers the opportunity to help promote your content items.

In this recipe, we look at how to implement a plugin that adds social bookmarking buttons for all the most popular sites.

Getting ready

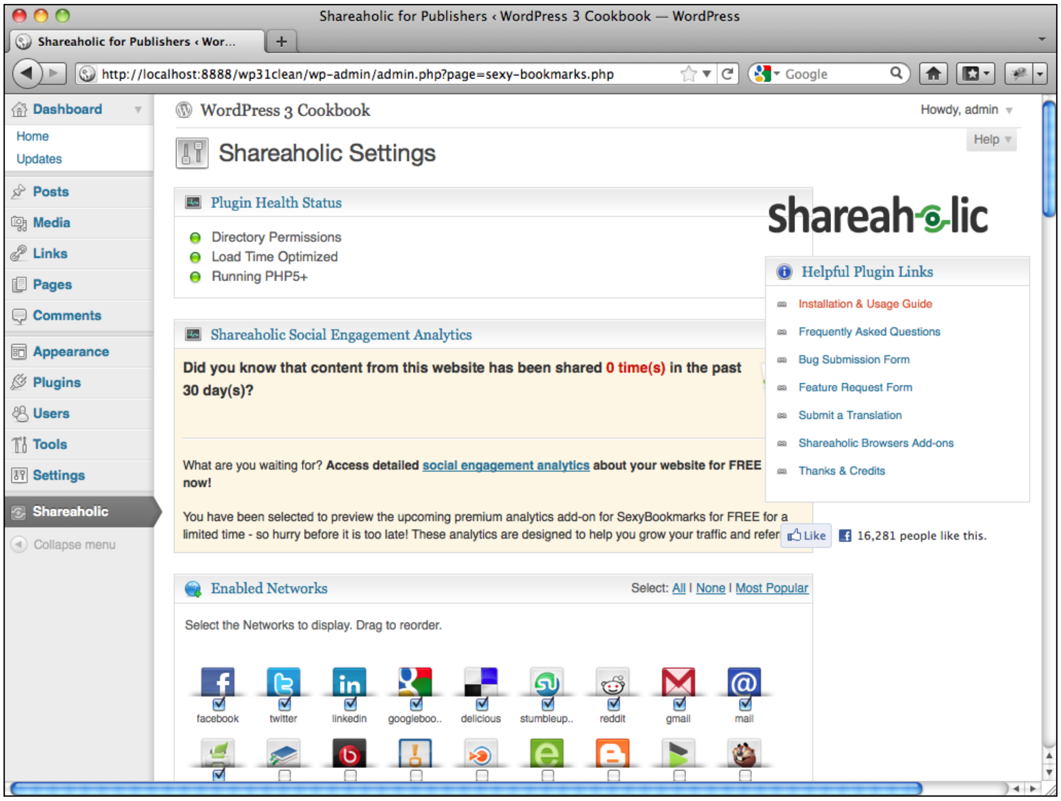
To execute this recipe, you will need to install the Sexy Bookmarks plugin (also known as "Shareaholic"). You will need to install this plugin before you can get started. Search for **Sexy Bookmarks** inside the **Add New Plugins** screen of your WordPress site. After you find it, click to install, and then activate it.

> You can learn more about the plugin by visiting the developer's site at `http://sexybookmarks.shareaholic.com/`

How to do it...

1. Log in to your WordPress **Dashboard**.
2. If you've just installed the Sexy Bookmarks plugin, you are likely to see a notice at the top of the page, advising you to configure the plugin. Click on the **Shareaholic** menu item.

3. On the **Shareaholic Settings** page, as shown in the next screenshot, you have a large number of options for configuring the appearance and functionality of this plugin. Select the options you wish to use.



Shareaholic for Publishers ‹ WordPress 3 Cookbook — WordPress

http://localhost:8888/wp31clean/wp-admin/admin.php?page=sexy-bookmarks.php

WordPress 3 Cookbook

Howdy, admin

Shareaholic Settings

Plugin Health Status

- Directory Permissions
- Load Time Optimized
- Running PHP5+

Shareaholic Social Engagement Analytics

Did you know that content from this website has been shared 0 time(s) in the past 30 day(s)?

What are you waiting for? Access detailed social engagement analytics about your website for FREE now!

You have been selected to preview the upcoming premium analytics add-on for SexyBookmarks for FREE for a limited time - so hurry before it is too late! These analytics are designed to help you grow your traffic and refer

Helpful Plugin Links

- Installation & Usage Guide
- Frequently Asked Questions
- Bug Submission Form
- Feature Request Form
- Submit a Translation
- Shareaholic Browsers Add-ons
- Thanks & Credits

16,281 people like this.

Enabled Networks

Select: All | None | Most Popular

Select the Networks to display. Drag to reorder.

Shareaholic for Publishers ‹ Wor...
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Shareaholic
Collapse menu
Help
shareaholic
Like
facebook
twitter
linkedin
googleboo..
delicious
stumbleup..
reddit
gmail
mail

4. Click on the **Save Changes** button.

How it works...

The plugin does all the work here. You need only configure it and you're ready to go!

There's more...

In addition to social bookmarking functionality, the Sexy Bookmarks plugin also supports a mail to a friend functionality and a print function. You can select these options from the lengthy list labeled **Enabled Networks**, as partially shown in the previous screenshot.

See also...

- *Bringing Facebook functionality into your site*

Aggregating RSS content

RSS feeds are common across the web and represent a great opportunity for you to enrich the content of your site and provide your visitors with related relevant content. If you are running a news site, RSS feeds are an invaluable way of providing more comprehensive coverage of your topic.

While the default WordPress system makes it easy to add individual feeds in widget positions, if you want to manage multiple feeds and produce the content in the main content areas of your pages, the default system is awkward to use. Fortunately there is a great plugin that enables you to aggregate and manage multiple RSS feeds. In this recipe, we introduce you to this handy plugin and show you how to complete a basic setup that will provide your site with RSS-sourced content items.

Getting ready

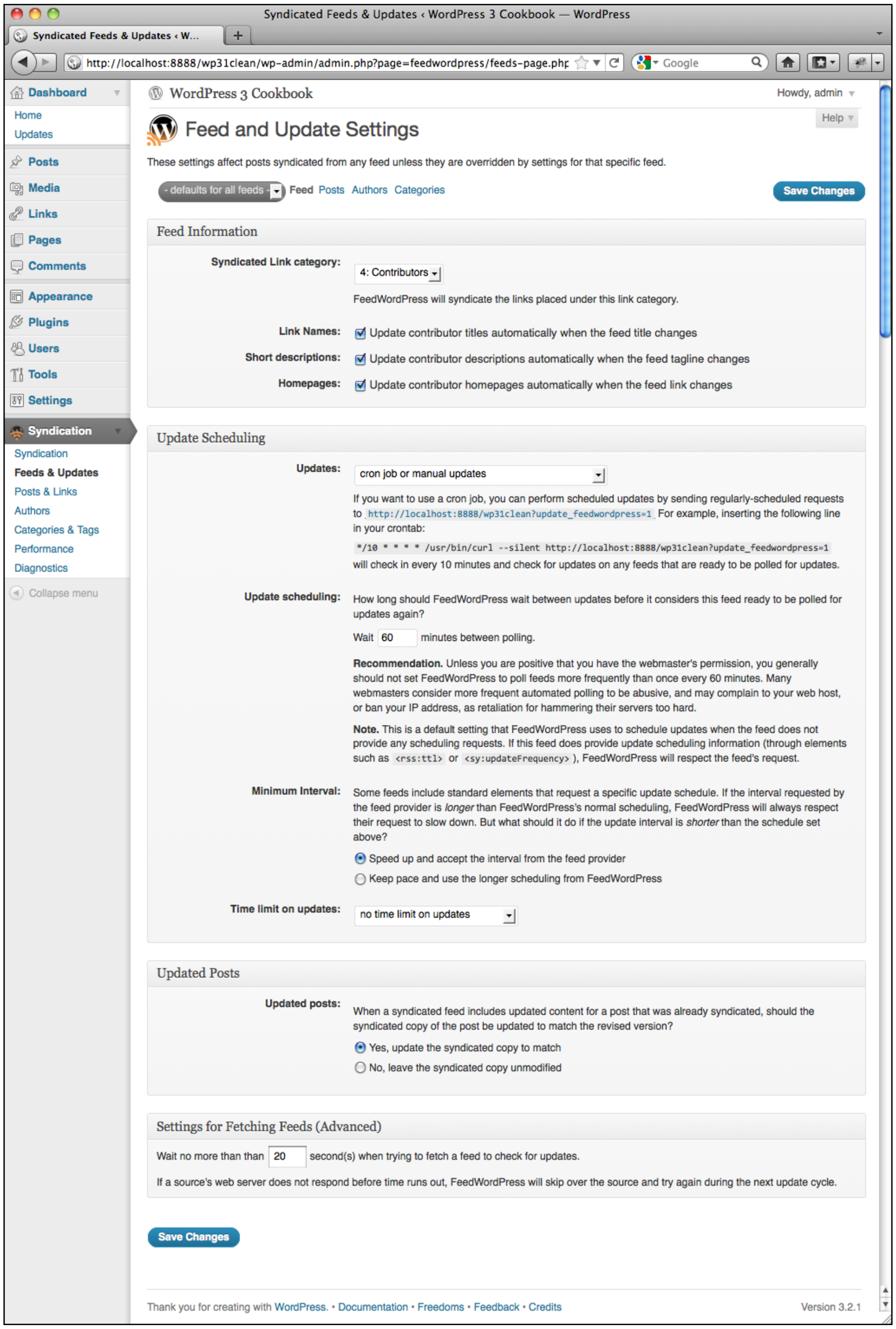
To execute this recipe, you will need to install the Feed WordPress plugin. You will need to install this plugin before you can get started. Search for **Feed Wordpress** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's site at `http://feedwordpress.radgeek.com/`

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Syndication** option on the admin menu.
3. On the page that loads, enter the URL of the first RSS feed you wish to aggregate.
4. Click on the **Add** button.
5. On the page that loads, check the options you wish to apply to this feed.
6. Click on the **Use this feed** button.
7. Repeat the process as needed to gather all the RSS feeds you wish to display.
8. Next, you need to schedule the updates. On the **Syndicated Sites Settings** page, click on the **Feed & Update Settings** link.

9. The **Feed and Update Settings** page, as shown in the next screenshot, provides all the options you need to schedule the updating of your feeds:


Syndicated Feeds & Updates ‹ WordPress 3 Cookbook — WordPress
Syndicated Feeds & Updates ‹ W...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=feedwordpress/feeds-page.php
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Syndication
Syndication
Feeds & Updates
Posts & Links
Authors
Categories & Tags
Performance
Diagnostics
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
- defaults for all feeds -
Feed
Posts
Authors
Categories
Save Changes

Feed and Update Settings

These settings affect posts syndicated from any feed unless they are overridden by settings for that specific feed.

Feed Information

Syndicated Link category: 4: Contributors

FeedWordPress will syndicate the links placed under this link category.

Link Names: ☑ Update contributor titles automatically when the feed title changes

Short descriptions: ☑ Update contributor descriptions automatically when the feed tagline changes

Homepages: ☑ Update contributor homepages automatically when the feed link changes

Update Scheduling

Updates: cron job or manual updates

If you want to use a cron job, you can perform scheduled updates by sending regularly-scheduled requests to `http://localhost:8888/wp31clean?update_feedwordpress=1` For example, inserting the following line in your crontab:

```
*/10 * * * * /usr/bin/curl --silent http://localhost:8888/wp31clean?update_feedwordpress=1
```

will check in every 10 minutes and check for updates on any feeds that are ready to be polled for updates.

Update scheduling: How long should FeedWordPress wait between updates before it considers this feed ready to be polled for updates again?

Wait 60 minutes between polling.

Recommendation. Unless you are positive that you have the webmaster's permission, you generally should not set FeedWordPress to poll feeds more frequently than once every 60 minutes. Many webmasters consider more frequent automated polling to be abusive, and may complain to your web host, or ban your IP address, as retaliation for hammering their servers too hard.

Note. This is a default setting that FeedWordPress uses to schedule updates when the feed does not provide any scheduling requests. If this feed does provide update scheduling information (through elements such as `<rss:ttl>` or `<sy:updateFrequency>`), FeedWordPress will respect the feed's request.

Minimum Interval: Some feeds include standard elements that request a specific update schedule. If the interval requested by the feed provider is *longer* than FeedWordPress's normal scheduling, FeedWordPress will always respect their request to slow down. But what should it do if the update interval is *shorter* than the schedule set above?

● Speed up and accept the interval from the feed provider
○ Keep pace and use the longer scheduling from FeedWordPress

Time limit on updates: no time limit on updates

Updated Posts

Updated posts: When a syndicated feed includes updated content for a post that was already syndicated, should the syndicated copy of the post be updated to match the revised version?

● Yes, update the syndicated copy to match
○ No, leave the syndicated copy unmodified

Settings for Fetching Feeds (Advanced)

Wait no more than than 20 second(s) when trying to fetch a feed to check for updates.

If a source's web server does not respond before time runs out, FeedWordPress will skip over the source and try again during the next update cycle.

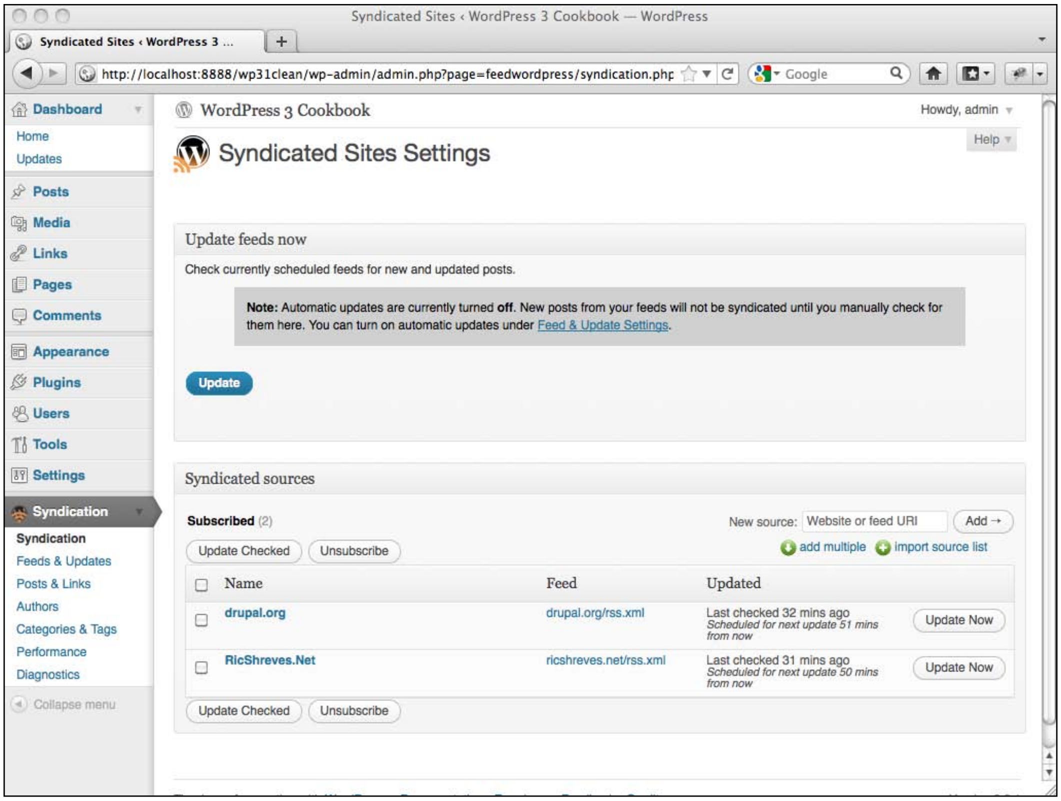
Save Changes
Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits
Version 3.2.1

10. Click on the **Save Changes** button.

> The **Updates** option enables you to select how the feeds will be updated. The options include manual updates, update before or after pages load, and cron. If you wish to use cron, you will have to take additional steps to set up a cron job on your server, as explain in the text immediately underneath the **Updates** heading, seen in the previous screenshot.

The plugin is ready to go; however, until you update the feeds to bring in the content items, there will be no content to display. To update your feeds for the first time, follow these steps:

1. Click on the **Syndication** option on the **Syndication** menu.
2. In the **Syndicated sources** section of the page, as shown in the next screenshot, click on the checkbox next to the name of all the feeds you wish to update:


Syndicated Sites ‹ WordPress 3 Cookbook — WordPress
Syndicated Sites ‹ WordPress 3 ...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=feedwordpress/syndication.php
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Syndication
Syndication
Feeds & Updates
Posts & Links
Authors
Categories & Tags
Performance
Diagnostics
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
Update
Add →
add multiple
import source list
Update Checked
Unsubscribe
Update Now

Syndicated Sites Settings

Update feeds now

Check currently scheduled feeds for new and updated posts.

Note: Automatic updates are currently turned **off**. New posts from your feeds will not be syndicated until you manually check for them here. You can turn on automatic updates under Feed & Update Settings.

Syndicated sources

Subscribed (2)

New source: Website or feed URI

☐ Name	Feed	Updated	
☐ **drupal.org**	drupal.org/rss.xml	Last checked 32 mins ago *Scheduled for next update 51 mins from now*	Update Now
☐ **RicShreves.Net**	ricshreves.net/rss.xml	Last checked 31 mins ago *Scheduled for next update 50 mins from now*	Update Now

3. Click on the **Update checked** button.

The system will check the feeds for new items and, where it finds them, will add them as posts in your site.

> You should always seek permission before you display content written by others as posts on your site.

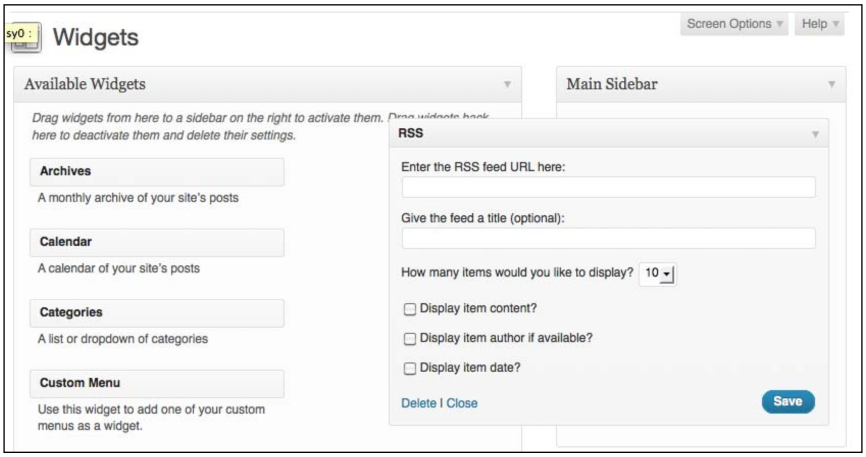
How it works...

This plugin allows you to query multiple RSS sources and to then bring the feed items into your site as posts. The multiple configuration options enable you to select how you want to display the posts and how to manage updating. There's no additional work required here on your part, unless you want to add the cron update option, as noted above.

There's more...

If you don't need to aggregate multiple RSS feeds, there is a simpler solution. The default WordPress system gives you a ready-to-use RSS widget specifically for this purpose. To display the output of a single RSS feed in a widget on your site, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** option on the main admin menu.
3. Click on the **Widget** option on the Appearance menu.
4. Click and drag the RSS widget on to the widget area where you want the output to appear.
5. Inside the new widget, enter the RSS URL, and set the other options you want, as shown in the following screenshot:



Widgets

Available Widgets

Drag widgets from here to a sidebar on the right to activate them. Drag widgets back here to deactivate them and delete their settings.

Archives
A monthly archive of your site's posts

Calendar
A calendar of your site's posts

Categories
A list or dropdown of categories

Custom Menu
Use this widget to add one of your custom menus as a widget.

Main Sidebar

RSS

Enter the RSS feed URL here:

Give the feed a title (optional):

How many items would you like to display? 10

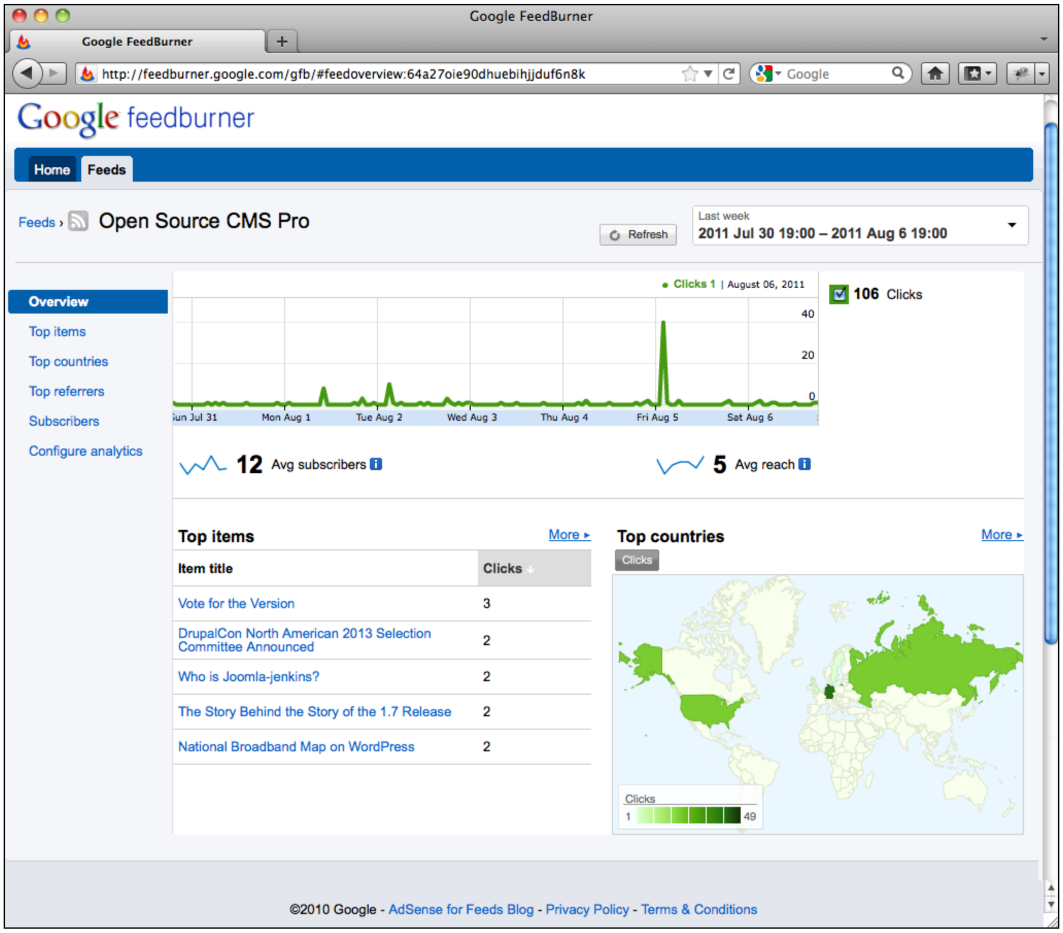
☐ Display item content?
☐ Display item author if available?
☐ Display item date?

6. Click on the **Save** button.

If you visit the front-end of your site, you should see the output from the RSS feed. This process can be repeated as many times as you like; the system will support multiple instances of the RSS widget.

Integrating Feedburner into your site

Feedburner is a service that allows you to keep a count of the people who have subscribed for your RSS feed. The Feedburner service is a great way to keep up with the RSS subscribers to your site and gain insight into what's working and what needs improvement. The following screenshot shows the Feedburner analytics page for a site:


Google FeedBurner
http://feedburner.google.com/gfb/#feedoverview:64a27oie90dhuebihjjduf6n8k
Google
Google feedburner
Home
Feeds
Feeds › Open Source CMS Pro
Refresh
Last week
2011 Jul 30 19:00 – 2011 Aug 6 19:00
Overview
Top items
Top countries
Top referrers
Subscribers
Configure analytics
● Clicks 1 | August 06, 2011
40
20
0
Sun Jul 31
Mon Aug 1
Tue Aug 2
Wed Aug 3
Thu Aug 4
Fri Aug 5
Sat Aug 6
☑ 106 Clicks
12 Avg subscribers
5 Avg reach
Top items
More ▸
Top countries
More ▸
Clicks
Clicks
1
49
©2010 Google - AdSense for Feeds Blog - Privacy Policy - Terms & Conditions

Item title	Clicks
Vote for the Version	3
DrupalCon North American 2013 Selection Committee Announced	2
Who is Joomla-jenkins?	2
The Story Behind the Story of the 1.7 Release	2
National Broadband Map on WordPress	2

Once you have created your Feedburner feed, you have to integrate it to your theme. In this recipe, we look at a plugin that helps you integrate the Feedburner feeds with your theme.

Getting ready

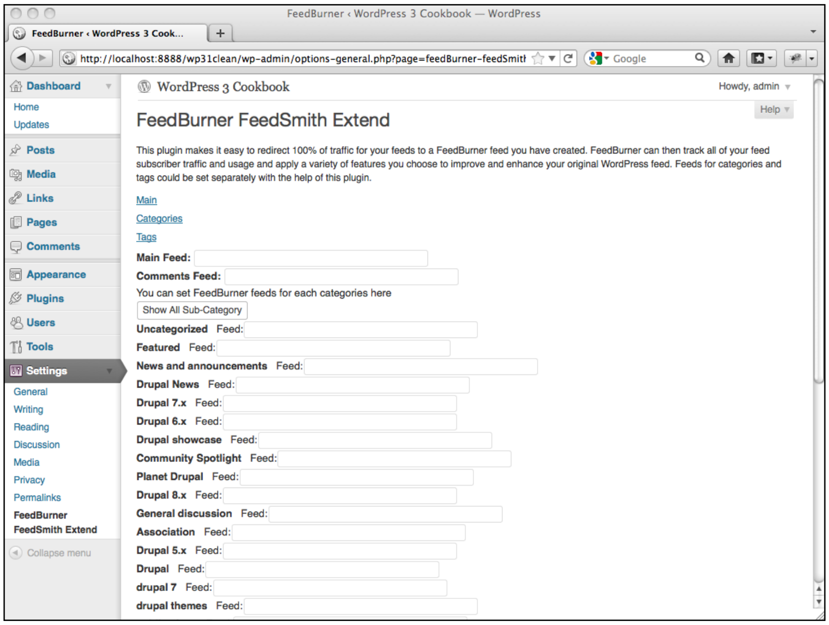
Before we can begin this recipe, you need to have a Feedburner account. Feedburner is a free service owned by Google. To set up your Feedburner account, visit `http://feedburner.google.com`

You will need to install the FeedBurner FeedSmith Extend plugin. You will need to install this plugin before you can get started. Search for **Feedburner Feedsmith Extend** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

How to do it...

Let's start with the manual way by carrying out the following steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the **FeedburnerFeedSmith Extend** option in the **Settings** menu.
4. Add your Feedburner URL to the field marked **Main Feed**, as you can see in the following screenshot:


FeedBurner ‹ WordPress 3 Cookbook — WordPress
FeedBurner ‹ WordPress 3 Cook...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=feedBurner-feedSmitl
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
FeedBurner FeedSmith Extend
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
Main
Categories
Tags
Show All Sub-Category

FeedBurner FeedSmith Extend

This plugin makes it easy to redirect 100% of traffic for your feeds to a FeedBurner feed you have created. FeedBurner can then track all of your feed subscriber traffic and usage and apply a variety of features you choose to improve and enhance your original WordPress feed. Feeds for categories and tags could be set separately with the help of this plugin.

Main Feed:
Comments Feed:
You can set FeedBurner feeds for each categories here
Uncategorized Feed:
Featured Feed:
News and announcements Feed:
Drupal News Feed:
Drupal 7.x Feed:
Drupal 6.x Feed:
Drupal showcase Feed:
Community Spotlight Feed:
Planet Drupal Feed:
Drupal 8.x Feed:
General discussion Feed:
Association Feed:
Drupal 5.x Feed:
Drupal Feed:
drupal 7 Feed:
drupal themes Feed:

5. Add any other feed URLs you want in the remaining fields.
6. Click on the **Save** button.

How it works...

The FeedSmith plugin simply redirects all requests for the standard WordPress RSS to the Feedburner URLs you set up in the configuration process.

Displaying a retweet button on your posts

Retweeting refers to the process of repeating a post on Twitter. Retweeting is also an effective method for publicizing your content items. By placing a retweet button on your content items, users can simply click the retweet button next to the content item and the system will automatically post it to Twitter, and log the retweet.

In this recipe we show you how to implement a plugin that makes it easy to add a retweet button to the content items on your site.

Getting ready

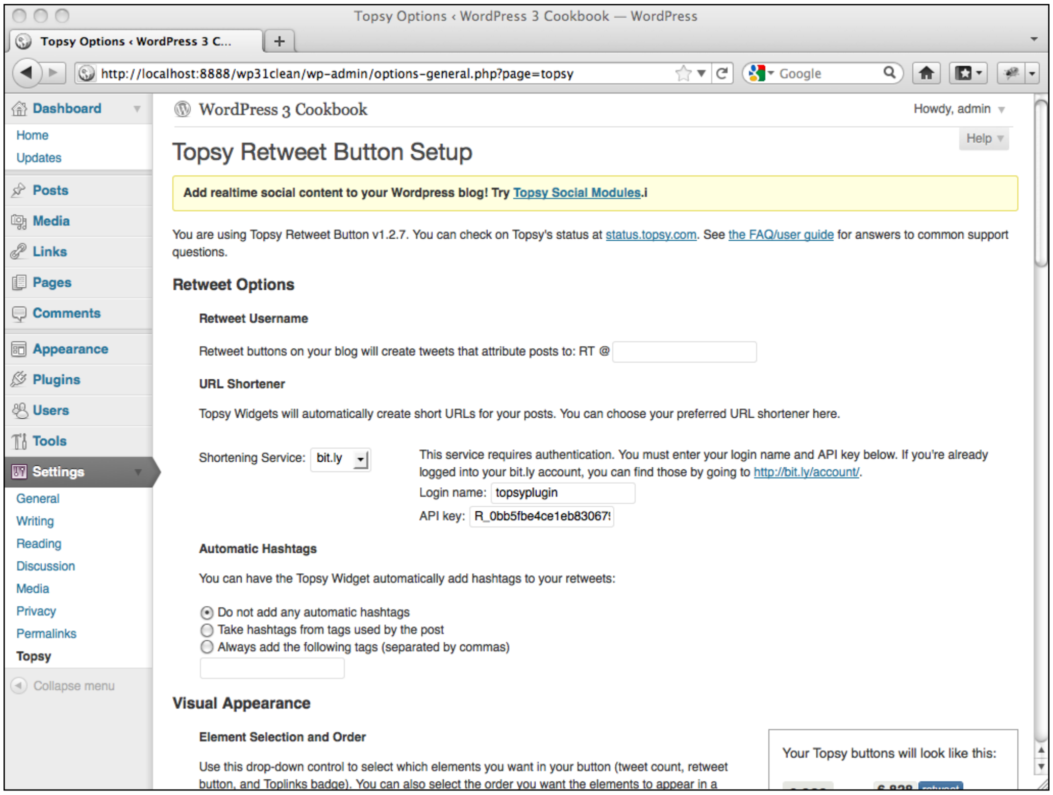
To execute this recipe, you will need to install the **TopsyRetweet Button** plugin. You will need to install this plugin before you can get started. Search for **TopsyRetweet** button inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's site at `http://corp.topsy.com/2009/10/13/wordpress-plugin-updated-version-0-9-1/`

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the **Topsy** link in the **Settings** menu.

4. On the **TopsyRetweet Button Setup** page, as shown in the next screenshot, you can select from a wide variety of options to customize how the post will be broadcast, whether the URL will be shortened (always a good idea on Twitter!), which categories will include the button, and options to control the buttons appearance and placement. Make your choices here, or simply use the defaults.


Topsy Options ‹ WordPress 3 Cookbook — WordPress
Topsy Options ‹ WordPress 3 C...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=topsy
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Topsy
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help

Topsy Retweet Button Setup

Add realtime social content to your Wordpress blog! Try Topsy Social Modules.i

You are using Topsy Retweet Button v1.2.7. You can check on Topsy's status at status.topsy.com. See the FAQ/user guide for answers to common support questions.

Retweet Options

Retweet Username

Retweet buttons on your blog will create tweets that attribute posts to: RT @

URL Shortener

Topsy Widgets will automatically create short URLs for your posts. You can choose your preferred URL shortener here.

Shortening Service: bit.ly

This service requires authentication. You must enter your login name and API key below. If you're already logged into your bit.ly account, you can find those by going to http://bit.ly/account/.

Login name: topsyplugin

API key: R_0bb5fbe4ce1eb83067

Automatic Hashtags

You can have the Topsy Widget automatically add hashtags to your retweets:

● Do not add any automatic hashtags
○ Take hashtags from tags used by the post
○ Always add the following tags (separated by commas)

Visual Appearance

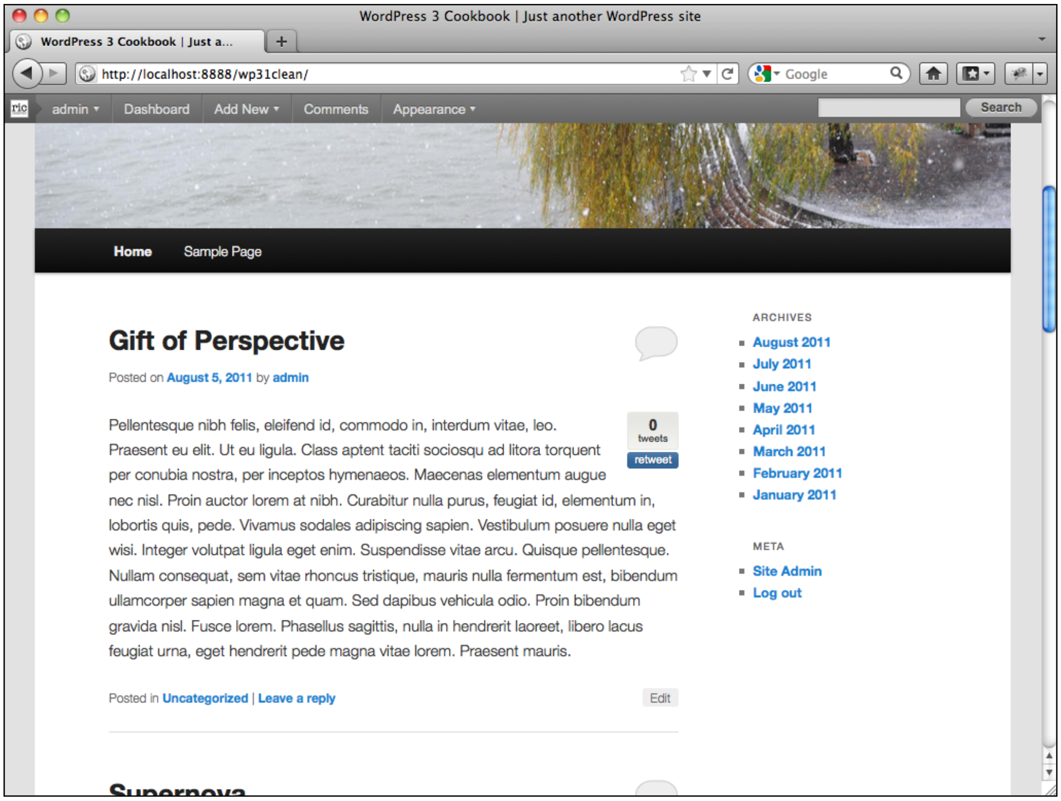
Element Selection and Order

Use this drop-down control to select which elements you want in your button (tweet count, retweet button, and Toplinks badge). You can also select the order you want the elements to appear in a

Your Topsy buttons will look like this:

5. Click on the **Save Changes** button.

The retweet button is now ready to go. Check the front-end of your site to verify the placement and styling; you should see something similar to the next screenshot. Give it a click and try it out.


WordPress 3 Cookbook | Just another WordPress site
WordPress 3 Cookbook | Just a...
http://localhost:8888/wp31clean/
Google
admin
Dashboard
Add New
Comments
Appearance
Search
Home
Sample Page
0
tweets
retweet
Edit

Gift of Perspective

Posted on August 5, 2011 by admin

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Maecenas elementum augue nec nisl. Proin auctor lorem at nibh. Curabitur nulla purus, feugiat id, elementum in, lobortis quis, pede. Vivamus sodales adipiscing sapien. Vestibulum posuere nulla eget wisi. Integer volutpat ligula eget enim. Suspendisse vitae arcu. Quisque pellentesque. Nullam consequat, sem vitae rhoncus tristique, mauris nulla fermentum est, bibendum ullamcorper sapien magna et quam. Sed dapibus vehicula odio. Proin bibendum gravida nisl. Fusce lorem. Phasellus sagittis, nulla in hendrerit laoreet, libero lacus feugiat urna, eget hendrerit pede magna vitae lorem. Praesent mauris.

Posted in Uncategorized | Leave a reply

ARCHIVES

- August 2011
- July 2011
- June 2011
- May 2011
- April 2011
- March 2011
- February 2011
- January 2011

META

- Site Admin
- Log out

How it works...

When a user clicks on the retweet button, they are prompted to log in to their Twitter account. Once they log in, the short message with the link to your content item is posted to Twitter through their account and is then viewable by anyone who follows that user (and to those who find the post by searching on Twitter).

Getting more comments with the Subscribe to Comments Reloaded plugin

Comments are very important on the web, because they encourage discussion. When leaving a comment on a post, it's tiresome to come back to the page repeatedly to see if anyone has replied or adding their own comments to the post. There is a solution: Add a subscription function to the comments.

In this recipe we look at implementing a plugin that will add a comments subscription functionality to your site.

Getting ready

To execute this recipe, you will need to install the Subscribe to Comments Reloaded plugin. You will need to install this plugin before you can get started. Search for **Subscribe to Comments Reloaded** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the **Subscribe to Comments** option in the **Settings** menu.
4. Click on the **Comment Form** tab at the top of the screen. This tab contains the options related to the visual appearance of the form and the messages that will be shown to your site visitors.
5. Select the options you desire.
6. Click on the **Save Changes** button.
7. You can also configure the **Management Page** options on the tab of the same name; this tab controls the availability and appearance of a management page for the subscribers to see all the comments to which they have subscribed and modify their subscriptions. The Management Page is enabled by default.
8. The **Notifications** tab lets you control what is sent out to subscribers in the notifications e-mails. The **Notifications** can be configured, but it is not necessary to make changes to this tab; the default settings will be fine for most people.
9. Finally, you should also check the **Options** tab and see if the default settings are sufficient for your needs.

How it works...

When someone leaves a comment on your blog, he or she can check the **Subscribetocomments** checkbox. If he or she checks it, the person will receive any other comment posted on the same article through his or her e-mail. This is particularly useful when debating with other readers, or when you have asked a question to the author.

There's more...

If you want, you can check out the subscriber list, see how many people subscribed to your blog posts, and manage subscriptions. To do all of this, log in to your WordPress **Dashboard** and go to **the Subscribe to Comments page under the Settings menu**.

From here, you can perform the following tasks:

- Search if a particular email address is subscribing to one or more of your posts
- Get e-mail addresses of your subscribers
- Get a list of most subscribed posts
- Remove subscribers
- Change e-mail address

Remove the nofollow attribute to motivate users to leave comments

By default, the links left in the comments have a `rel="nofollow"` attribute automatically added by WordPress. According to Wikipedia, `nofollow` is an HTML attribute value used to instruct some search engines that a hyperlink should not influence the link's target ranking in the search engine's index. It is intended to reduce the effectiveness of certain types of search engine spam, thereby improving the quality of search engine results and preventing spamdexing from occurring.

In other words, any link pointing to a site provides a potential boost to the site's relevancy ranking. If a link has a `rel="nofollow"` attribute, it will not provide any boost to the linked site.

In order to reward commentators it is a good idea to get rid of this `rel="nofollow"` attribute on links. This way, your commentators will gain some benefit when they leave comments on your site.

In this recipe we implement a plugin to selectively remove the nofollow links and thereby provide an incentive to people to post comments on your site.

Getting ready

To execute this recipe, you will need to install the Remove Comment NoFollow plugin. You will need to install this plugin before you can get started. Search for **Remove Comment nofollow** inside the **Add New Plugins** screen of your WordPress site. After you find it, click to install, and then activate it.

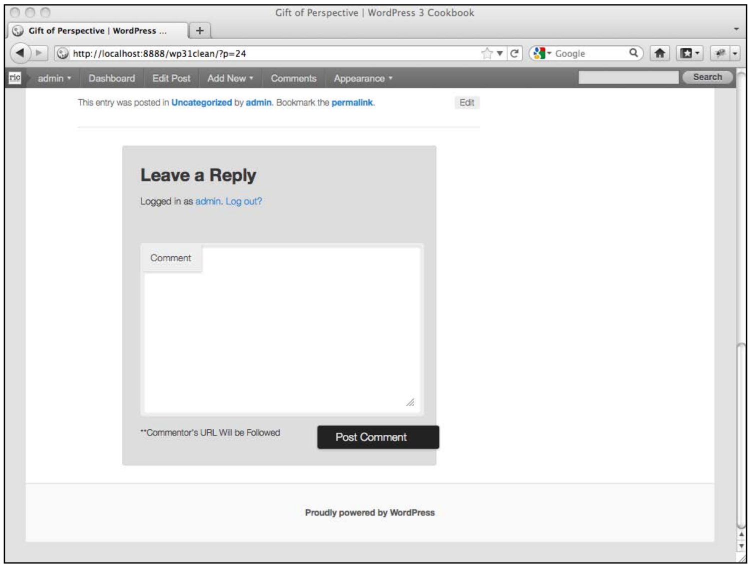
> You can learn more about the plugin by visiting the developer's site at `http://myhub.gumz-ex-press.com/remove-comment-nofollow-plugin`

How to do it...

There is only one configuration option available for this plugin. To access that option, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the **GMZ Comment Follow** option in the **Settings** menu.
4. Enter the message you want to display on the comments form.
5. Click on the **Save Changes** button.

That's all there is to it. As you can see in the following screenshot, the message you enter appears on the comment to help motivate users to post comments on your site:


Gift of Perspective | WordPress 3 Cookbook
Gift of Perspective | WordPress ...
http://localhost:8888/wp31clean/?p=24
Google
admin
Dashboard
Edit Post
Add New
Comments
Appearance
Search
Edit
Comment
Post Comment

This entry was posted in Uncategorized by admin. Bookmark the permalink.

Leave a Reply

Logged in as admin. Log out?

**Commentor's URL Will be Followed

Proudly powered by WordPress

How it works...

The plugin in this recipe provides an alternative to hacking the core files to remove the nofollow attribute. As we want to always avoid hacking the core files, the plugin provides us with an easy solution.

Once the nofollow instruction is removed, the search engines will credit the target website with an additional inbound link, thereby providing the site with an incremental boost in its relevancy rankings.

Provide recognition to your top contributors

Another good way to reward commentators is to provide them with some recognition. The Top Contributors widget enables you to display a list of the people who provide the most comments or articles for your site. Placing the widget on your site is a good way to not only motivate people to be active on your site, but also a good way to provide direct links to the content generated by your most valuable users.

Getting ready

To execute this recipe, you will need to install the Top Contributors plugin. You will need to install this plugin before you can get started. Search for **Top Contributors** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's site at `http://justmyecho.com/2010/07/top-contributors-plugin-wordpress/`

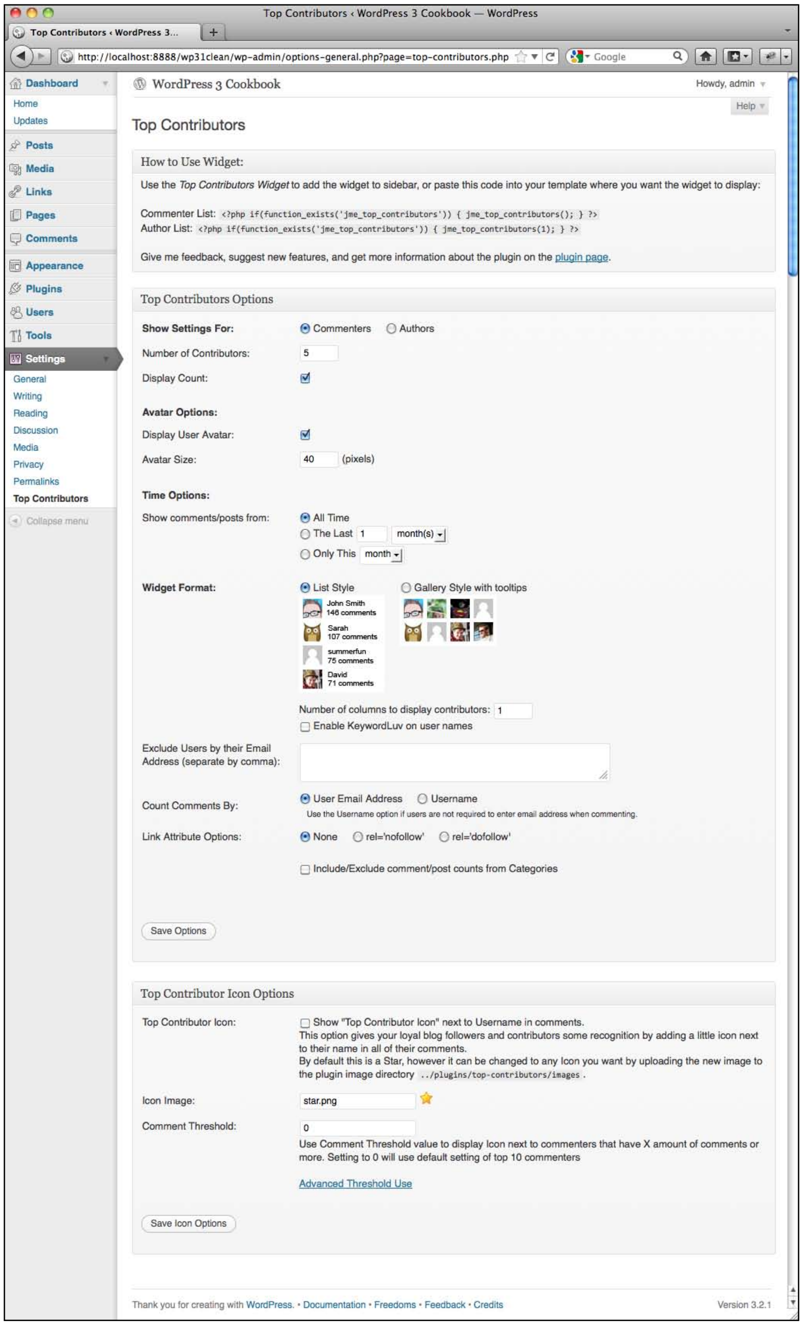
How to do it...

There are two parts to this recipe: First we must configure the plugin, then we must set up the widget.

Carry out the following steps to configure the Top Contributors plugin:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click the **Top Contributors** link in the **Settings** menu.

4. Select from the options displayed, as shown in the following screenshot:


Top Contributors ‹ WordPress 3 Cookbook — WordPress
Top Contributors ‹ WordPress 3...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=top-contributors.php
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Top Contributors
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
John Smith
146 comments
Sarah
107 comments
summerfun
75 comments
David
71 comments
Save Options
Save Icon Options
Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits
Version 3.2.1

Top Contributors

How to Use Widget:

Use the *Top Contributors Widget* to add the widget to sidebar, or paste this code into your template where you want the widget to display:

Commenter List: `<?php if(function_exists('jme_top_contributors')) { jme_top_contributors(); } ?>`
Author List: `<?php if(function_exists('jme_top_contributors')) { jme_top_contributors(1); } ?>`

Give me feedback, suggest new features, and get more information about the plugin on the plugin page.

Top Contributors Options

Show Settings For: ● Commenters ○ Authors
Number of Contributors: 5
Display Count: ☑

Avatar Options:
Display User Avatar: ☑
Avatar Size: 40 (pixels)

Time Options:
Show comments/posts from: ● All Time
○ The Last 1 month(s)
○ Only This month

Widget Format: ● List Style ○ Gallery Style with tooltips
Number of columns to display contributors: 1
☐ Enable KeywordLuv on user names

Exclude Users by their Email Address (separate by comma):

Count Comments By: ● User Email Address ○ Username
Use the Username option if users are not required to enter email address when commenting.

Link Attribute Options: ● None ○ rel='nofollow' ○ rel='dofollow'

☐ Include/Exclude comment/post counts from Categories

Top Contributor Icon Options

Top Contributor Icon: ☐ Show "Top Contributor Icon" next to Username in comments.
This option gives your loyal blog followers and contributors some recognition by adding a little icon next to their name in all of their comments.
By default this is a Star, however it can be changed to any Icon you want by uploading the new image to the plugin image directory `../plugins/top-contributors/images` .

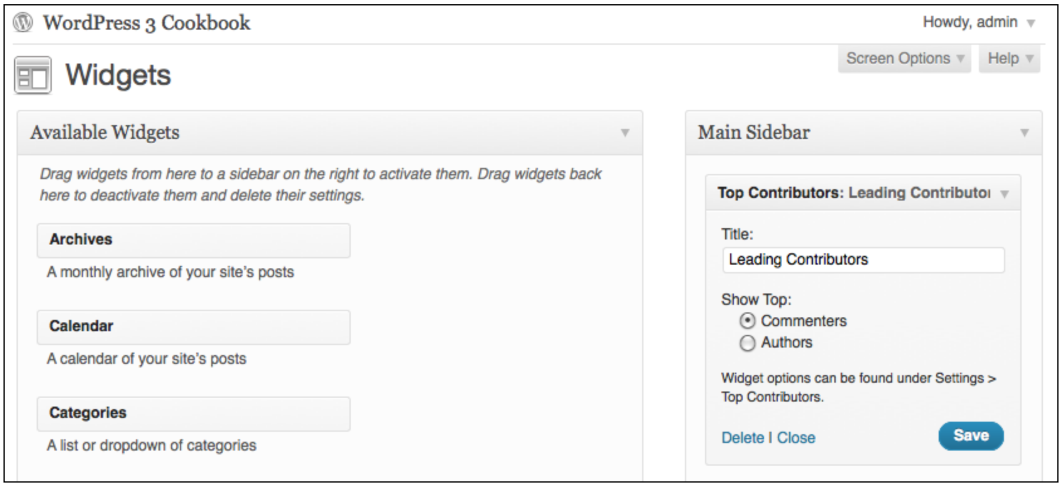
Icon Image: star.png
Comment Threshold: 0
Use Comment Threshold value to display Icon next to commenters that have X amount of comments or more. Setting to 0 will use default setting of top 10 commenters

Advanced Threshold Use

5. Click on the relevant **Save** button.

Now that the plugin is configured, you still need to set up the widget to display the list of top contributors on your site. To set up the widget, follow these steps:

1. Click on the **Appearance** menu.
2. Click on the **Widgets** option.
3. Click and drag the **Top Contributors** widget to the sidebar where you wish it to appear.
4. In the widget options, shown in the following image, give the widget a name and select whether it will show top commentators or the top authors.
5. Click on the **Save** button.


WordPress 3 Cookbook
Howdy, admin
Screen Options
Help
Widgets
Available Widgets
Archives
Calendar
Categories
Main Sidebar
Top Contributors: Leading Contributo
Delete | Close
Save

Drag widgets from here to a sidebar on the right to activate them. Drag widgets back here to deactivate them and delete their settings.

A monthly archive of your site's posts

A calendar of your site's posts

A list or dropdown of categories

Title: Leading Contributors

Show Top:
● Commenters
○ Authors

Widget options can be found under Settings > Top Contributors.

The widget should now be visible on your site.

How it works...

The plugin does all the work here; it looks into the database to obtain the information on all the users. Your configuration options simply tell it how to filter the users and which to display.

Displaying author-related information on posts

It's always good for the reader to know the author of the article that they're currently reading. It's even better if they can learn some extra information about the author, such as his website, a short bio, and so on. While many modern themes offer the option to display author info on posts, if your theme does not, you may want to consider adding this information to your posts, as it benefits both your site visitors and your authors.

In this recipe, you'll learn how to edit your `single.php` theme file to automatically retrieve the author-related information, and display it at the top of the page.

> Note that many themes today offer the option to display author info, photos, or avatars. This recipe, therefore, is intended for themes that lack this capacity.

Getting ready

All you need to execute this recipe is your favorite text editor and access to the theme files of your WordPress installation.

How to do it...

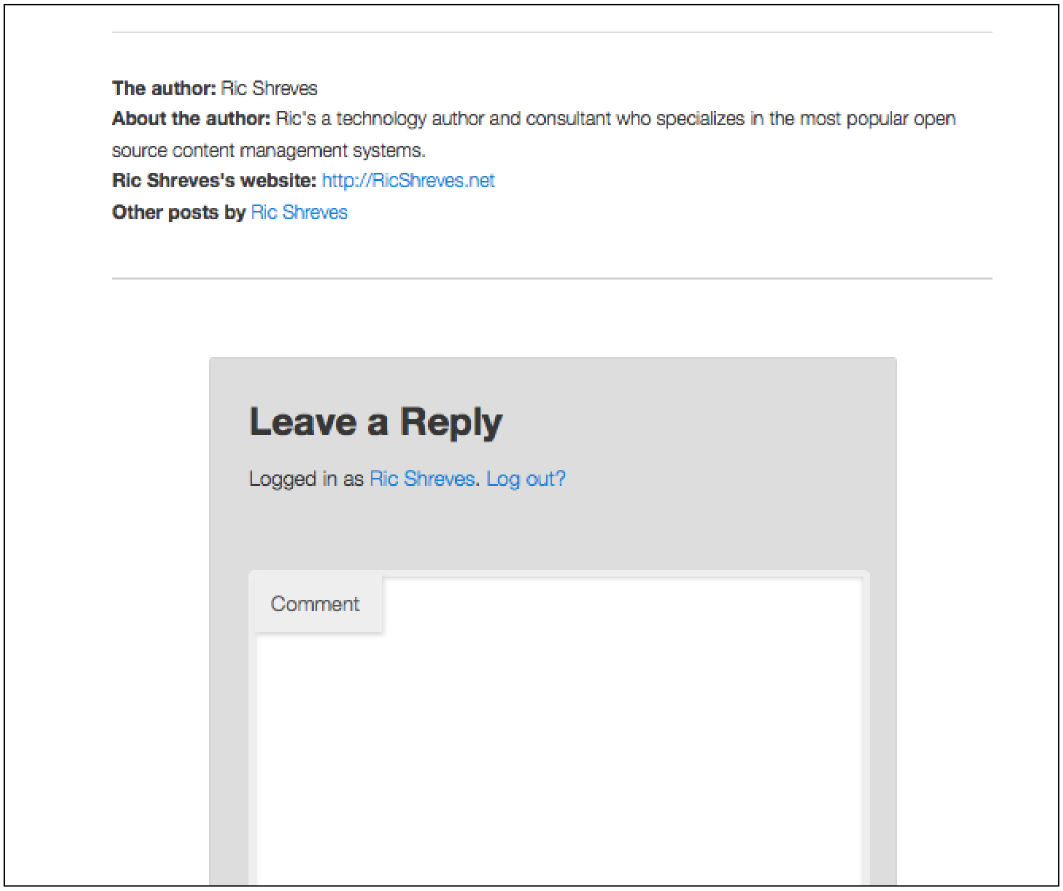
Once you have made sure that your authors have successfully filled in their information, you can start coding by carrying out the following steps:

1. Access your WordPress installation.
2. Find the directory containing your active theme's files.
3. Open the `single.php` file for editing.
4. Find the loop, and paste the following inside it:

```
<div id="author-info">
   <strong>The author:</strong><?phpthe_author();?><br />
   <strong>About the author:</strong><?phpthe_author_
meta('description'); ?><br />
   <strong><?phpthe_author();?>'s website:</strong><a
href="<?phpthe_author_meta('user_url'); ?>"><?phpthe_author_url();
?></a><br />
   <strong>Other posts by</strong><?phpthe_author_posts_link(); ?>
   </div>
<br />
<hr />
<br />
<!--/author-info-->
```

5. Save the file.

If you access your site now you will see the following appended to the post pages:



The author: Ric Shreves
About the author: Ric's a technology author and consultant who specializes in the most popular open source content management systems.
Ric Shreves's website: http://RicShreves.net
Other posts by Ric Shreves

Leave a Reply

Logged in as Ric Shreves. Log out?

How it works...

WordPress provides a dozen author-related template tags, which are an easy way to retrieve information that is entered by authors in their profile. The tags rely upon the data entered into the user's profile. Users can either enter that themselves, or it can be added by an administrator with privileges sufficient to edit the users.

There's more...

The code in this recipe relies on the use of template tags, primarily `the_author_meta()`, which includes a number of parameters that provide the output in this recipe. Here's a listing of the more useful parameters for this key template tag:

Parameter	Output
▸ `description`	The author's description (Bio)
▸ `user_firstname`	The author's first name
▸ `user_lastname`	The author's last name
▸ `nickname`	The author's nickname
▸ `author_url`	The author's website URL
▸ `aim`	The author's AIM screen name
▸ `yim`	The author's Yahoo! Messenger email

Displaying the author's avatar on posts

WordPress provides site administrators with multiple options for adding user images. By default, the system supports several choices, including the popular Gravatar service.

In this recipe, we'll show you how to display the author's avatar on posts. This recipe is a good complement to the previous recipe showing how to display author info on a post.

> Note that many themes today offer the option to display author info, photos, or avatars. This recipe, therefore, is intended for themes that lack this capacity.

Getting ready

All you need to complete this recipe is a text editor and access to your WordPress theme files on your server.

How to do it...

1. Access the WordPress installation on your server.
2. Open for editing the `single.php` file of your active theme.
3. Locate the loop.
4. Add the following line of code inside the loop, where you want the avatar to appear:

   ```
   <?php echo get_avatar( get_the_author_meta( 'user_email' ) ); ?>
   ```

5. Save the file.

How it works...

This recipe uses `the_author_meta()` function we saw earlier in this chapter. In this case, we use the function to retrieve the avatar image associated with the author and display it on the page.

Allowing multiple authors on posts

In a multi-author blog, sometimes an author starts a post and another one finishes it. Or, perhaps, two or more contributors share their ideas and create a great article together.

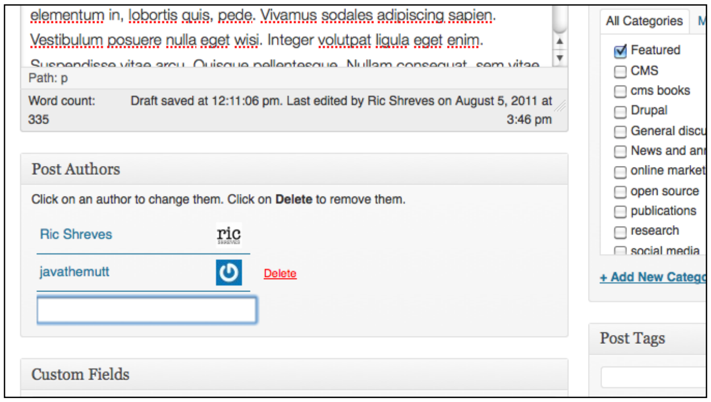
Unfortunately, by default, WordPress allows only one author per post. Although this is good for most blogs, it can quickly become very frustrating for contributors on a multi-author blog. Just imagine that two authors have worked together to write a post, but only one can be rewarded; that's not a very comfortable situation for either of them.

Luckily, there is a WordPress plugin that allows your site to assign more than one author to a post. The plugin is named Co-Authors Plus and in this recipe we look at implementing it on your site.

How to do it...

This plugin requires no additional configuration – simply install it and activate it and you are ready to go. To add additional authors to a post, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Open for editing a content item.
3. Scroll down to the area just below the editing window and you will see a new dialogue box on the page: **Post Authors**.
4. To add another author, start typing in the empty field you can see in the following image:



elementum in, lobortis quis, pede. Vivamus sodales adipiscing sapien. Vestibulum posuere nulla eget wisi. Integer volutpat ligula eget enim.

Path: p

Word count: 335

Draft saved at 12:11:06 pm. Last edited by Ric Shreves on August 5, 2011 at 3:46 pm

Post Authors

Click on an author to change them. Click on **Delete** to remove them.

Ric Shreves

javathemutt Delete

5. The system will attempt to autocomplete the author field for you as you type. As soon as you find the name you want, click on it and the system will automatically add it to the list of authors for the post.
6. Click the **Update** button to save your changes.

While the plugin requires no additional configuration, if you wish to display the additional authors on the front-end of the website, you do need to modify your template files. To add display co-authors on your site, follow these steps:

1. Access your WordPress installation on the server.
2. Open the directory for the active theme.
3. Open the file (or files) containing the loop.
4. Find the loop.
5. Insert inside the loop the following:

```
<?php if(function_exists('coauthors_posts_links'))
    coauthors_posts_links();
else
the_author_posts_link(); ?>
```

6. Save the file.

If you visit the front end of your website, you should see co-authors listed on the relevant items.

> Note that the example shows just one way of achieving the output of the co-authors' names, in this case with a link to a page containing all the posts of each author. In the next section we look at other options you can use. Note also that the placement of the code inside the loop will affect the appearance. Typically you will want to place this code immediately beneath `the_title()` to maintain standard placement.

How it works...

The plugin simply associates registered users with posts. The additional template tags the plugin provides, discussed below, give you the flexibility to control the output.

There's more...

The plugin provides you with a set of co-author template tags, which function in a similar fashion as `the_author()` tags discussed earlier in this chapter. The new template tags are:

- `coauthors()`
- `coauthors_posts_links()`

- `coauthors_firstnames()`
- `coauthors_lastnames()`
- `coauthors_nicknames()`
- `coauthors_links()`
- `coauthors_IDs()`

> For more insight into the options presented by the plugin's template tags, see `http://wordpress.org/extend/plugins/co-authors-plus/other_notes/`

Displaying a list of all of the authors

If your site features content from multiple contributors, then you might want to think about including a page listing all the authors, with links to their contributions. Not only is such a page potentially useful to your site visitors, but also your authors tend to appreciate it and additionally, it can increase your internal link density, thereby giving you a search engine advantage.

In this recipe, we look at creating a page listing all your site's authors, with links to their contributions.

Getting ready

To achieve this recipe you will want create a page template to hold the output of the author listing; though this is not mandatory, it makes things a bit simpler and enables you to style the page effectively and without complication.

Once you have your page template created, all you need to complete this recipe is a text editor and access to the WordPress installation on your server.

> Creating custom page templates is covered in *Chapter 2, Installing and Customizing Themes*.

How to do it...

There are two parts to this recipe. First, we must add a line of code to one of your page templates, and then we need to create and publish a page using the template:

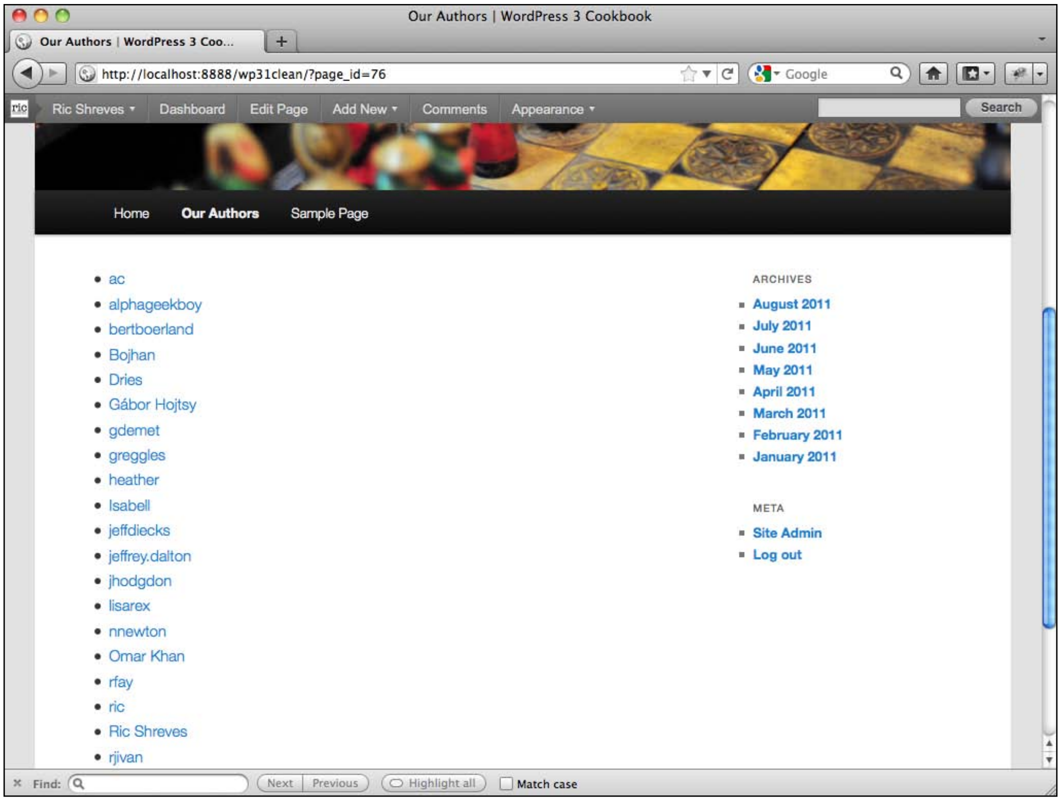
1. Access your WordPress installation on your server.
2. Go to the directory containing the active theme.

3. Open the page template you created for this recipe (or any existing page template).
4. Add the following line of code where you want the list of authors to appear:

```
<?phpwp_list_authors(); ?>
```

5. Save the template file.
6. Log in to your WordPress **Dashboard**.
7. Click on the **Pages** menu.
8. Click on the **Add New** link in the **Pages** menu.
9. Give your new page a title.
10. Select form the **Page Attributes** section in the right column the **Template** that holds the code above.
11. Publish the page.

If you now visit your site, you can visit the page created, where you should see something similar to the following screenshot:


Our Authors | WordPress 3 Cookbook
Our Authors | WordPress 3 Coo...
http://localhost:8888/wp31clean/?page_id=76
Google
Ric Shreves
Dashboard
Edit Page
Add New
Comments
Appearance
Search
Home
Our Authors
Sample Page
Find:
Next
Previous
Highlight all
Match case

- ac
- alphageekboy
- bertboerland
- Bojhan
- Dries
- Gábor Hojtsy
- gdemet
- greggles
- heather
- Isabell
- jeffdiecks
- jeffrey.dalton
- jhodgdon
- lisarex
- nnewton
- Omar Khan
- rfay
- ric
- Ric Shreves
- rjivan

ARCHIVES

- August 2011
- July 2011
- June 2011
- May 2011
- April 2011
- March 2011
- February 2011
- January 2011

META

- Site Admin
- Log out

How it works...

To display the list of your authors, you only need the `wp_list_authors()` function. This function can be inserted anywhere into your theme files. Once called, `wp_list_authors()` executes a database query to get the list of authors. The function can be controlled with parameters.

There's more...

The `wp_list_authors()` function offers several parameters, as shown in the following table:

Parameter	**Description**
`echo`	Display the results.
`exclude_admin`	Exclude the admin account from the list.
`feed`	Display a link to the author's RSS feed.
`feed_image`	Display the graphic from the author's RSS feed.
`feed_type`	The type of feed.
`hide_empty`	Do not display authors with no posts.
`html`	Determines whether the list of items is in html or plain text.
`orderby`	Sort the results.
`number`	The maximum number of users to display.
`optioncount`	Display the number of posts published by each user.
`order`	Order, ascending or descending.
`show_fullname`	Display the first and last name of the users.
`style`	Style in which to display the list.

To learn more about the `wp_list_authors()` function, visit `http://codex.wordpress.org/Function_Reference/wp_list_authors`

Creating community with BuddyPress

In this chapter we've looked at a number of recipes that add little bits of functionality designed to improve interactivity and encourage community. There is, however, one plugin that is built to help you create a full-featured community site. The BuddyPress plugin provides turnkey social networking functionality. Among the features included in this plugin:

- Forum
- Groups
- Member directory
- Activity stream
- Custom user profile creation

Of course, in addition to those features, you still have available the full range of standard WordPress functions and it's all extensible too, with a large number of plugins you can install to further customize BuddyPress to your needs.

It's a complex package, but very well done. BuddyPress is not your typical plugin; it literally transforms how your site handles users and the way that they interact with content and with each other. Be certain you need the richness of BuddyPress before you install it, as it represents a significant change in the nature of your site.

In this recipe we look at the basic setup needed to get you up and running quickly.

> To get a better idea of what this plugin is capable of, visit the Showcase page on the BuddyPress site at `http://buddypress.org/showcase/`

Getting ready

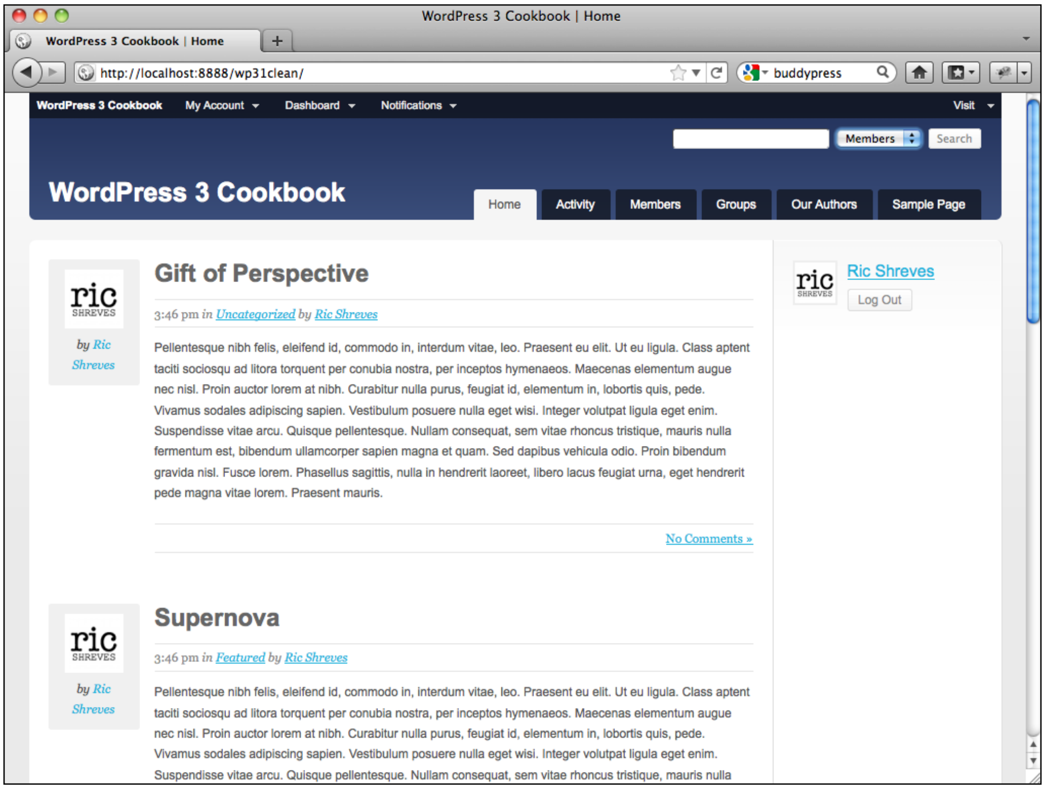
To execute this recipe, you will need to install the BuddyPress plugin. You will need to install this plugin before you can get started. Search for **BuddyPress** inside the Add New Plugins screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's site at `http://buddypress.org`

How to do it...

BuddyPress is a complex plugin and requires a bit of configuration if you want to get the most out of it. Follow these steps to get up and running quickly:

1. Log in to your WordPress **Dashboard**.
2. If you've just installed and activated **BuddyPress**, you may see a warning message at the top of the **Dashboard**, advising you to adjust your permalinks settings. **BuddyPress** will not work with the default permalinks settings. If you see a warning message, follow the given link and change the permalinks settings to anything other than the default option.
3. Click the **BuddyPress** option on the admin menu.
4. On the page that loads, you will likely see a message prompting you to select a **BuddyPress** compatible theme. If you see that message, click on the link labeled **activate a BuddyPress compatible theme**. Clicking on that link will take you to the themes page where you need to activate the **BuddyPress** theme. Once you have done that, return to the **BuddyPress Settings** page by clicking on the **BuddyPress** link on the admin menu. The default **BuddyPress** theme can be seen in the following screenshot:


WordPress 3 Cookbook | Home
WordPress 3 Cookbook | Home
http://localhost:8888/wp31clean/
buddypress
WordPress 3 Cookbook
My Account
Dashboard
Notifications
Visit
Members
Search
WordPress 3 Cookbook
Home
Activity
Members
Groups
Our Authors
Sample Page
ric SHREVES
by Ric Shreves
Ric Shreves
Log Out

Gift of Perspective

3:46 pm *in* Uncategorized *by* Ric Shreves

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Maecenas elementum augue nec nisl. Proin auctor lorem at nibh. Curabitur nulla purus, feugiat id, elementum in, lobortis quis, pede. Vivamus sodales adipiscing sapien. Vestibulum posuere nulla eget wisi. Integer volutpat ligula eget enim. Suspendisse vitae arcu. Quisque pellentesque. Nullam consequat, sem vitae rhoncus tristique, mauris nulla fermentum est, bibendum ullamcorper sapien magna et quam. Sed dapibus vehicula odio. Proin bibendum gravida nisl. Fusce lorem. Phasellus sagittis, nulla in hendrerit laoreet, libero lacus feugiat urna, eget hendrerit pede magna vitae lorem. Praesent mauris.

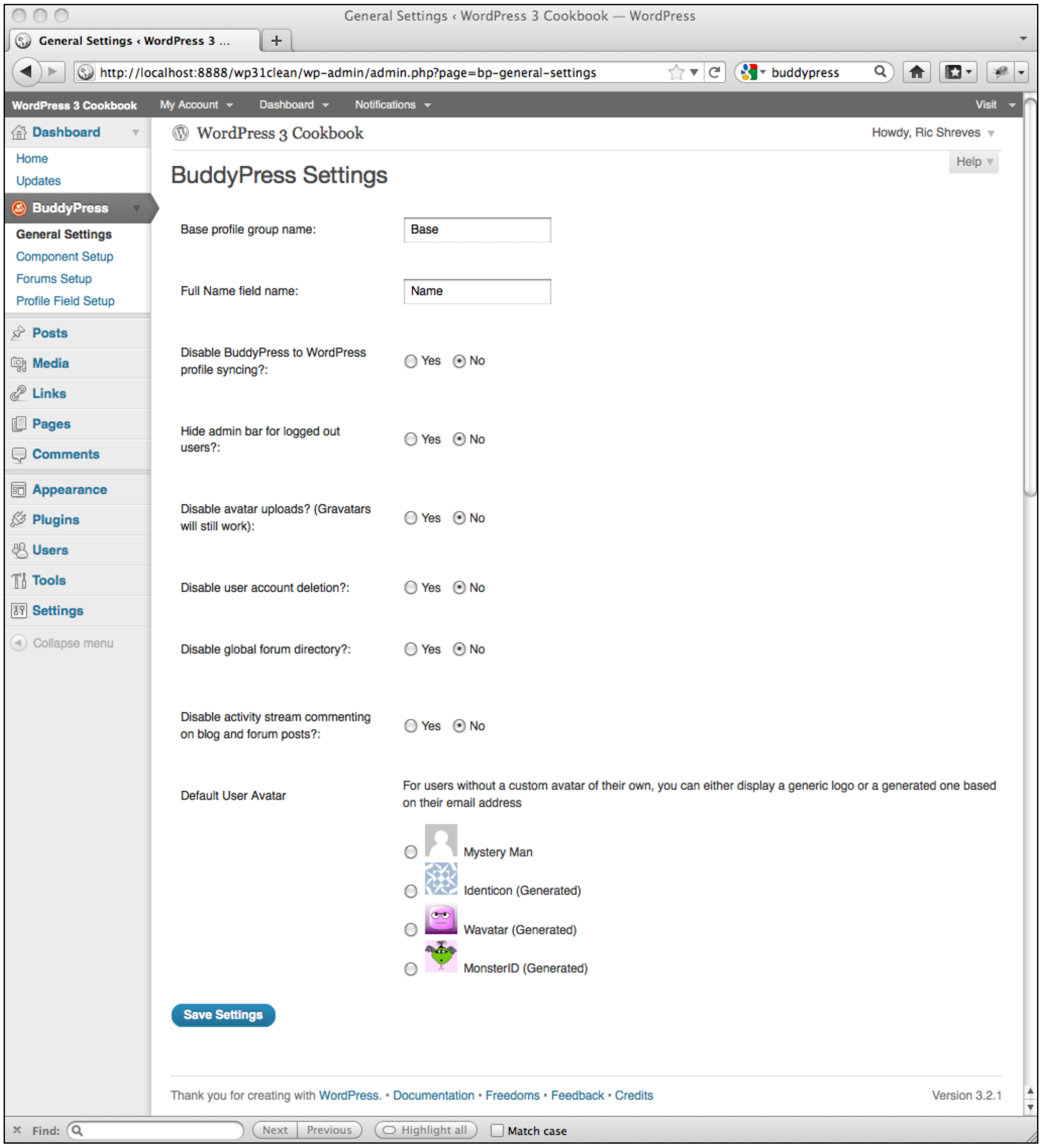
No Comments »

Supernova

3:46 pm *in* Featured *by* Ric Shreves

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Maecenas elementum augue nec nisl. Proin auctor lorem at nibh. Curabitur nulla purus, feugiat id, elementum in, lobortis quis, pede. Vivamus sodales adipiscing sapien. Vestibulum posuere nulla eget wisi. Integer volutpat ligula eget enim. Suspendisse vitae arcu. Quisque pellentesque. Nullam consequat, sem vitae rhoncus tristique, mauris nulla

5. The **General Settings** options, seen in the following image, establish some basic behaviors for the system. The default values are likely to be OK for most sites, but if you need to make adjustments, do so, then click on the **Save Settings** button.


General Settings ‹ WordPress 3 Cookbook — WordPress
General Settings ‹ WordPress 3 ...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=bp-general-settings
buddypress
WordPress 3 Cookbook
My Account
Dashboard
Notifications
Visit
Dashboard
Home
Updates
BuddyPress
General Settings
Component Setup
Forums Setup
Profile Field Setup
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Collapse menu
WordPress 3 Cookbook
Howdy, Ric Shreves
Help
Save Settings
Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits
Version 3.2.1
Find:
Next
Previous
Highlight all
Match case

BuddyPress Settings

Base profile group name: Base

Full Name field name: Name

Disable BuddyPress to WordPress profile syncing?: ○ Yes ● No

Hide admin bar for logged out users?: ○ Yes ● No

Disable avatar uploads? (Gravatars will still work): ○ Yes ● No

Disable user account deletion?: ○ Yes ● No

Disable global forum directory?: ○ Yes ● No

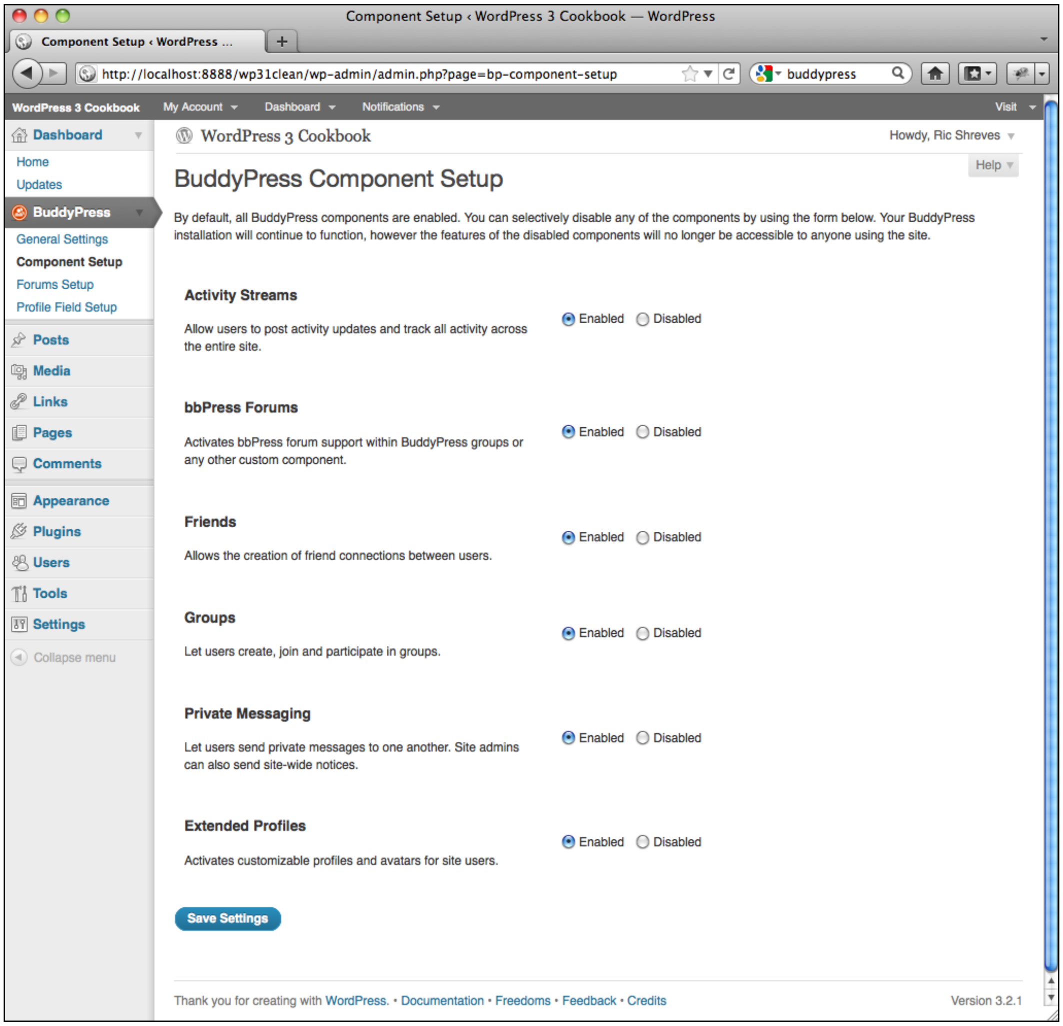
Disable activity stream commenting on blog and forum posts?: ○ Yes ● No

Default User Avatar

For users without a custom avatar of their own, you can either display a generic logo or a generated one based on their email address

○ Mystery Man
○ Identicon (Generated)
○ Wavatar (Generated)
○ MonsterID (Generated)

6. Click the **Component Settings** link on the **BuddyPress** menu. Select the components you wish to use from the list, seen in the screenshot below. By default, all components are active. Note that if you enable the **bbPress Forums** option, you will need to complete the **Forums Setup**, in the next step.


Component Setup ‹ WordPress 3 Cookbook — WordPress
Component Setup ‹ WordPress ...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=bp-component-setup
buddypress
WordPress 3 Cookbook
My Account
Dashboard
Notifications
Visit
Dashboard
Home
Updates
BuddyPress
General Settings
Component Setup
Forums Setup
Profile Field Setup
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Collapse menu
WordPress 3 Cookbook
Howdy, Ric Shreves
Help
Save Settings
Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits
Version 3.2.1

BuddyPress Component Setup

By default, all BuddyPress components are enabled. You can selectively disable any of the components by using the form below. Your BuddyPress installation will continue to function, however the features of the disabled components will no longer be accessible to anyone using the site.

Activity Streams
Allow users to post activity updates and track all activity across the entire site.
● Enabled ○ Disabled

bbPress Forums
Activates bbPress forum support within BuddyPress groups or any other custom component.
● Enabled ○ Disabled

Friends
Allows the creation of friend connections between users.
● Enabled ○ Disabled

Groups
Let users create, join and participate in groups.
● Enabled ○ Disabled

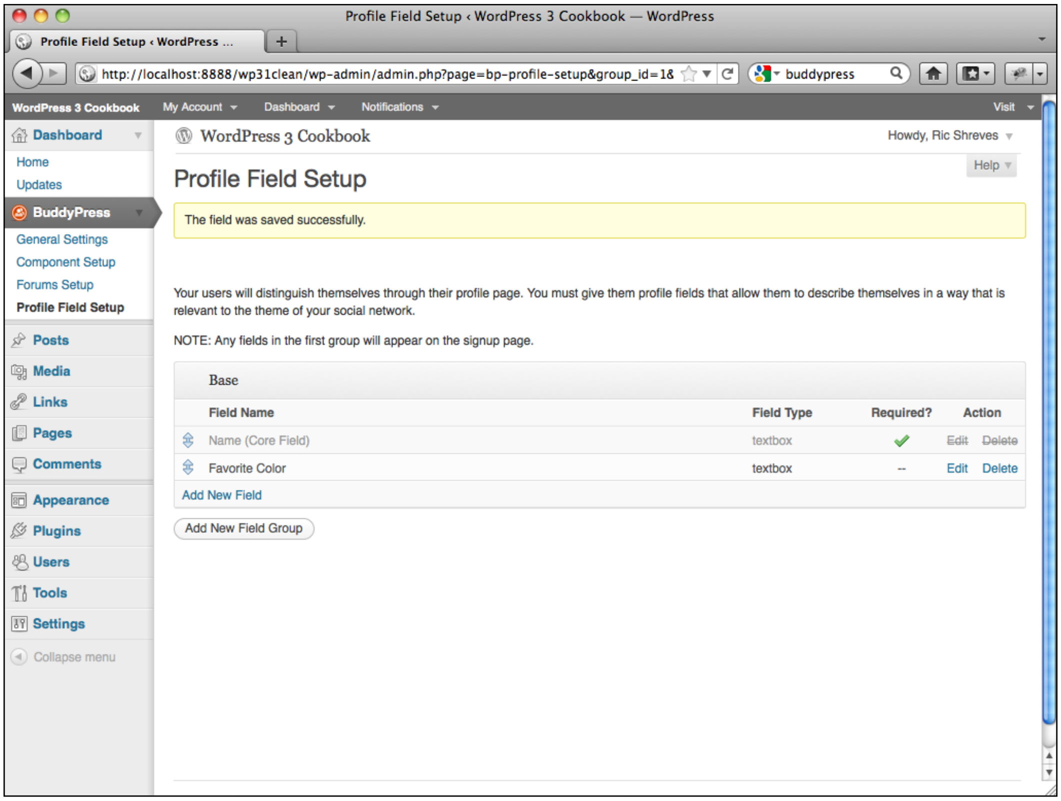
Private Messaging
Let users send private messages to one another. Site admins can also send site-wide notices.
● Enabled ○ Disabled

Extended Profiles
Activates customizable profiles and avatars for site users.
● Enabled ○ Disabled

7. If you enabled the **bbPress Forums** option on the previous screen, click on the **Forums Setup** option on the **BuddyPress** menu.

8. The screen contains two buttons: If you have an existing bbPress installation, select the second. If you need to set up bbPress, click the button labeled **Set up a new bbPress installation**. The system will then automatically set up the forum and add a link to your main nav.

9. The final step in the initial configuration is add any custom profile fields you require, if any. If you wish to add fields to the user profiles, then click on the **Profile Field Setup** link under the **BuddyPress** menu.
10. Click on the **Add New Field** button, as shown in the following screenshot:


Profile Field Setup ‹ WordPress 3 Cookbook — WordPress
Profile Field Setup ‹ WordPress ...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=bp-profile-setup&group_id=1&
buddypress
WordPress 3 Cookbook
My Account
Dashboard
Notifications
Visit
Dashboard
Home
Updates
BuddyPress
General Settings
Component Setup
Forums Setup
Profile Field Setup
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
Collapse menu
WordPress 3 Cookbook
Howdy, Ric Shreves
Help
Profile Field Setup
Add New Field
Add New Field Group

The field was saved successfully.

Your users will distinguish themselves through their profile page. You must give them profile fields that allow them to describe themselves in a way that is relevant to the theme of your social network.

NOTE: Any fields in the first group will appear on the signup page.

Base

Field Name	Field Type	Required?	Action
Name (Core Field)	textbox	✓	~~Edit~~ ~~Delete~~
Favorite Color	textbox	--	Edit Delete

11. The **Field Title** is required, but the **Description** is optional. Also select whether the new field will be required for users and the **Field Type**.
12. Click on the **Save** button.

The BuddyPress plugin is now ready to go. Visit the front-end of the site to explore the options.

How it works...

Though there is a lot going on, the BuddyPress plugin does all the work in this recipe.

See also...

- *Adding a forum into your site*

Adding a simple gallery to your site

If you've ever wanted to add a photo gallery to your WordPress site, you'll be glad to know that there is a gallery functionality built into the system. In this recipe we look at using the default gallery feature inside a post or a page; in the latter portions of this recipe, we dive into adding the gallery directly into a template file.

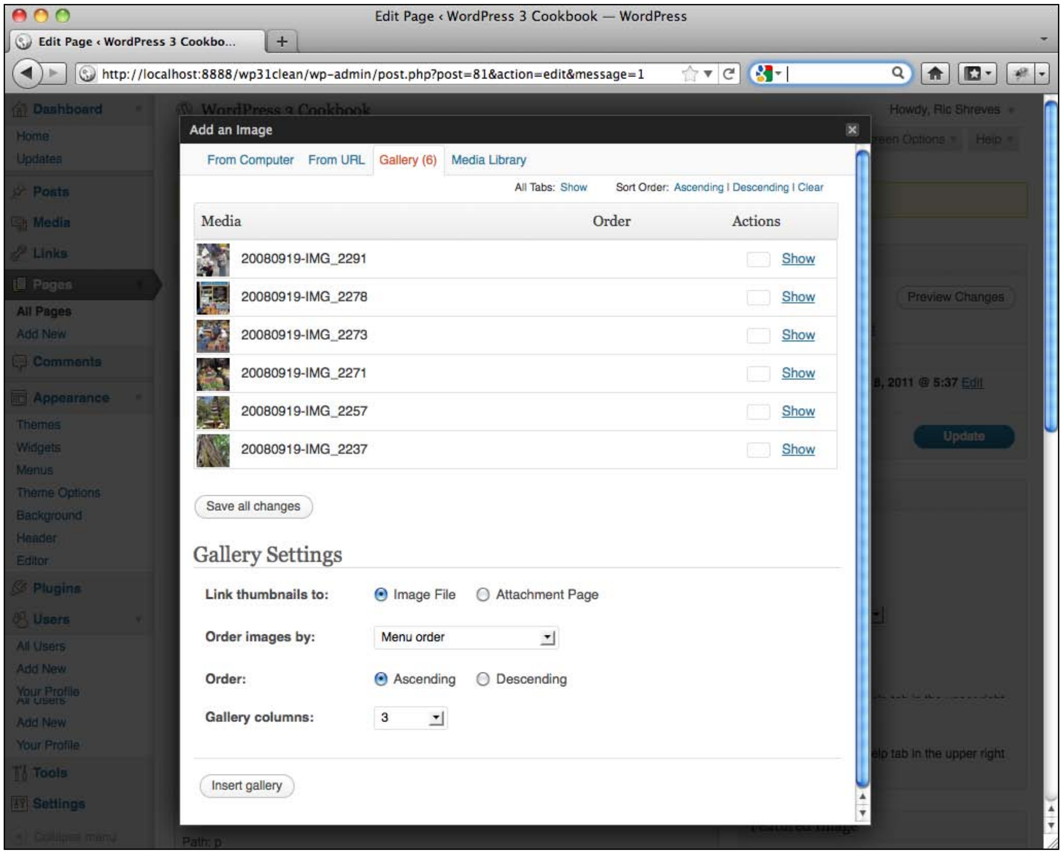
Getting ready

You don't need anything special to complete this recipe, only access to your server and your favorite code editor.

How to do it...

Let's create a new page and use it to hold a photo gallery. Follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Pages** link on the admin nav.
3. Click on the option **Add New**.
4. Name the new page **The Gallery**.
5. Click on the **Media** icon, next to the **Upload/Insert** label.
6. Upload the images you want to appear in your gallery. As you upload each, select **Save All Changes**, but do not insert the photo into the page.
7. On the **Gallery** tab, shown in the next screenshot, select the **Gallery Settings** you wish to use.


Edit Page ‹ WordPress 3 Cookbook — WordPress
http://localhost:8888/wp31clean/wp-admin/post.php?post=81&action=edit&message=1
Add an Image
From Computer
From URL
Gallery (6)
Media Library
All Tabs: Show
Sort Order: Ascending | Descending | Clear
Save all changes
Insert gallery

Media	Order	Actions
20080919-IMG_2291	☐	Show
20080919-IMG_2278	☐	Show
20080919-IMG_2273	☐	Show
20080919-IMG_2271	☐	Show
20080919-IMG_2257	☐	Show
20080919-IMG_2237	☐	Show

Gallery Settings

Link thumbnails to: ● Image File ○ Attachment Page

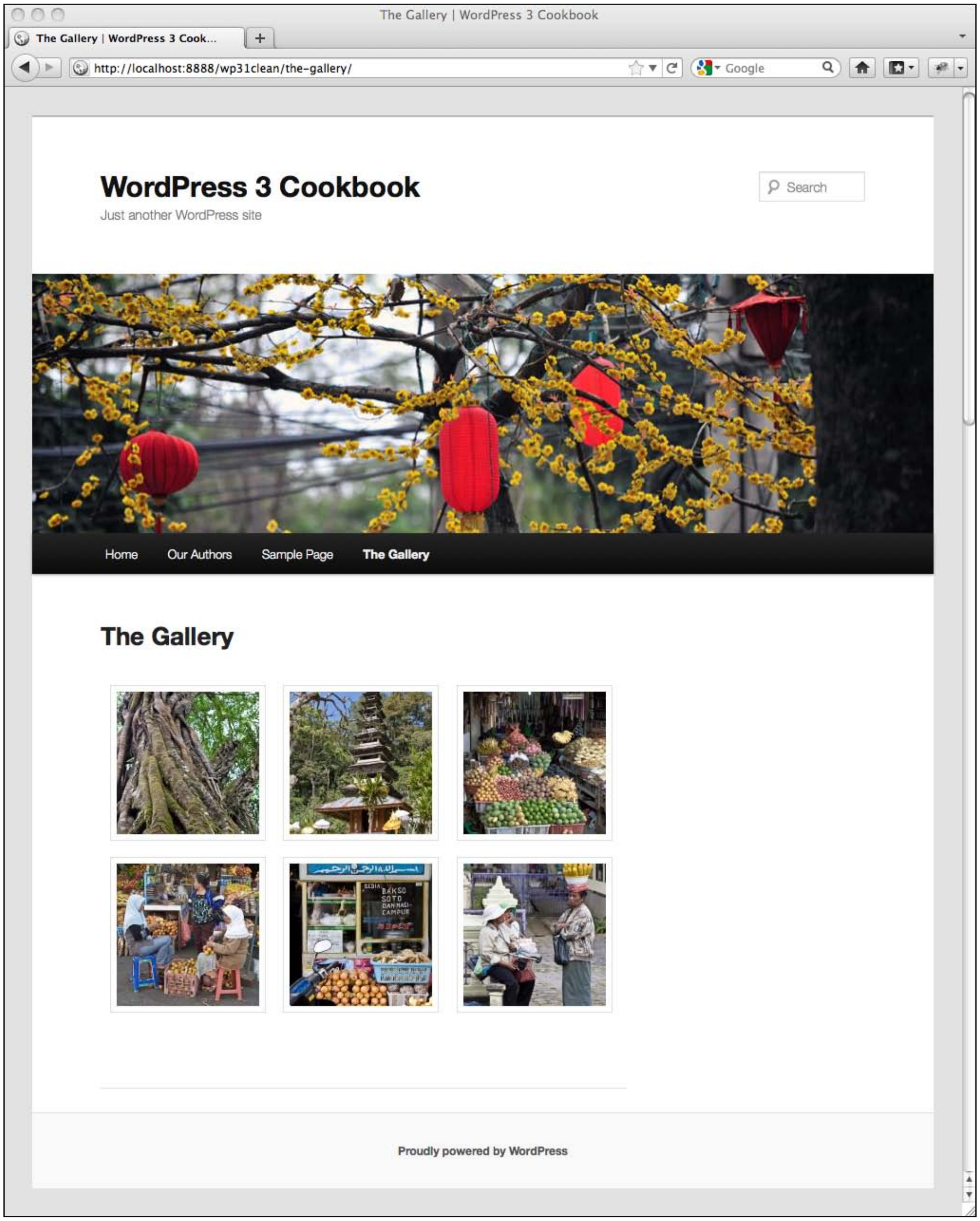
Order images by: Menu order

Order: ● Ascending ○ Descending

Gallery columns: 3

8. Click on the **Insert gallery** button.
9. **Publish** the page.

If you visit your site, and click on the link to the new page named **The Gallery**, you should see something similar to what is shown in the following screenshot:



The Gallery | WordPress 3 Cookbook
The Gallery | WordPress 3 Cook...
http://localhost:8888/wp31clean/the-gallery/
Google
WordPress 3 Cookbook
Just another WordPress site
Search
Home
Our Authors
Sample Page
The Gallery
The Gallery
Proudly powered by WordPress

How it works...

The gallery functionality is built into WordPress; you simply need to tell the system what images to use and which page or post you wish to use to hold the gallery. As you will see in the next section, you also have the option to use a shortcode.

There's more...

The gallery feature can also be inserted into content items or templates using the `[gallery]` shortcode. To use the code in a content item, first create the gallery by uploading images, then switch to the HTML editor for the content item and type the shortcode where you want the thumbnails to appear.

The shortcode can also be used inside templates, but the syntax is slightly different. To place a gallery inside a template, put this line of code where you want the thumbnails to appear:

```
<?php echo do_shortcode('[gallery]'); ?>
```

The gallery shortcode also includes a number of optional parameters. The following table shows the options available:

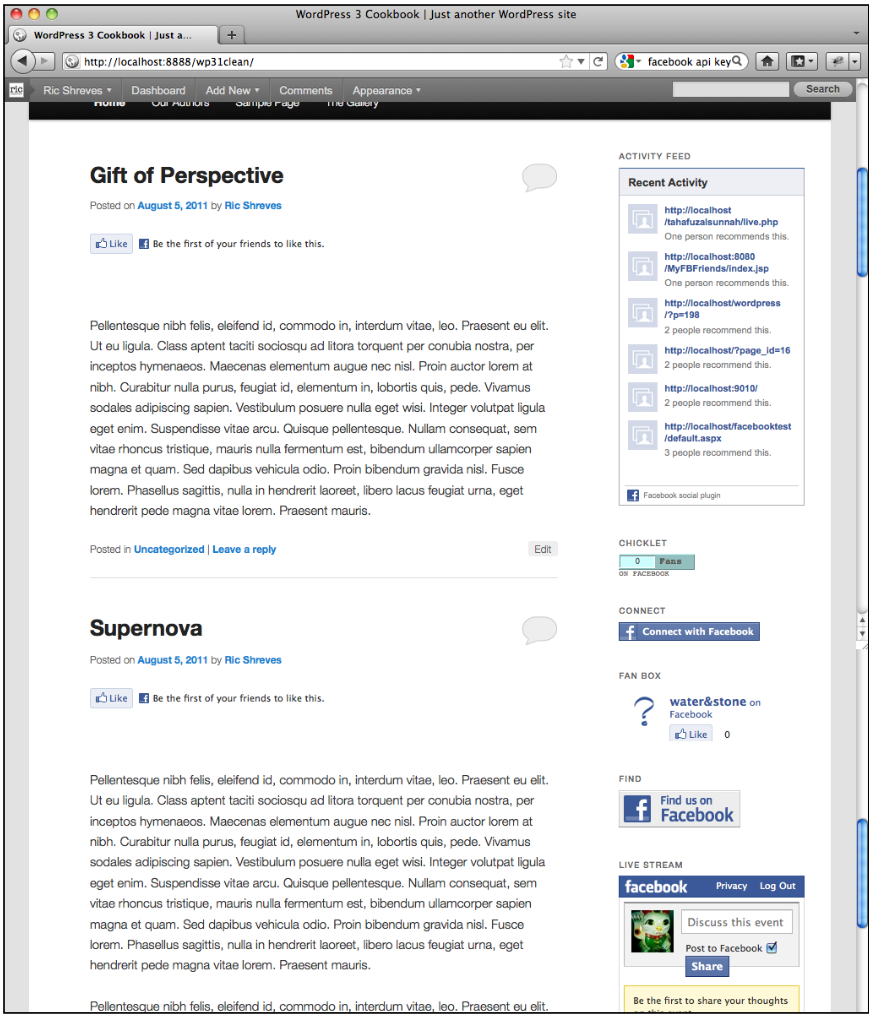
Parameter	**Description**
`Columns`	Number of columns for the thumbnails
`Link`	How each image will be linked
`Orderby`	How to order the images
`Size`	The size of the images to display
`Id`	Instructs the gallery to display the images from a specific post ID
`Include`	Include specific images
`Exclude`	Exclude specific images

> You can learn more about the WordPress gallery shortcode by visiting the WordPress Codex at `http://codex.wordpress.org/Gallery_Shortcode`

Bringing Facebook functionality into your site

The Facebook API offers website owners access to a variety of features. By connecting with the API, you can display widgets of Facebook content and user activity, allow users to log in using their Facebook I.D., or let them post your content to their Facebook account.

In this recipe we implement the Simple Facebook Connect plugin, which gives you access to a wide assortment of Facebook features that you can use to enhance your site. The following screenshot shows just some of the functions supported by the plugin (though obviously you'd never want to display them like this on your site!):


WordPress 3 Cookbook | Just another WordPress site
WordPress 3 Cookbook | Just a...
http://localhost:8888/wp31clean/
facebook api key
Ric Shreves
Dashboard
Add New
Comments
Appearance
Search
Home
Our Authors
Sample Page
The Gallery
Like
Edit
ACTIVITY FEED
Recent Activity
Facebook social plugin
CHICKLET
0 Fans
ON FACEBOOK
CONNECT
Connect with Facebook
FAN BOX
water&stone on Facebook
Like 0
FIND
Find us on Facebook
LIVE STREAM
facebook
Privacy
Log Out
Discuss this event
Post to Facebook ☑
Share

Gift of Perspective

Posted on August 5, 2011 by Ric Shreves

Be the first of your friends to like this.

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Maecenas elementum augue nec nisl. Proin auctor lorem at nibh. Curabitur nulla purus, feugiat id, elementum in, lobortis quis, pede. Vivamus sodales adipiscing sapien. Vestibulum posuere nulla eget wisi. Integer volutpat ligula eget enim. Suspendisse vitae arcu. Quisque pellentesque. Nullam consequat, sem vitae rhoncus tristique, mauris nulla fermentum est, bibendum ullamcorper sapien magna et quam. Sed dapibus vehicula odio. Proin bibendum gravida nisl. Fusce lorem. Phasellus sagittis, nulla in hendrerit laoreet, libero lacus feugiat urna, eget hendrerit pede magna vitae lorem. Praesent mauris.

Posted in Uncategorized | Leave a reply

Supernova

Posted on August 5, 2011 by Ric Shreves

Be the first of your friends to like this.

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Maecenas elementum augue nec nisl. Proin auctor lorem at nibh. Curabitur nulla purus, feugiat id, elementum in, lobortis quis, pede. Vivamus sodales adipiscing sapien. Vestibulum posuere nulla eget wisi. Integer volutpat ligula eget enim. Suspendisse vitae arcu. Quisque pellentesque. Nullam consequat, sem vitae rhoncus tristique, mauris nulla fermentum est, bibendum ullamcorper sapien magna et quam. Sed dapibus vehicula odio. Proin bibendum gravida nisl. Fusce lorem. Phasellus sagittis, nulla in hendrerit laoreet, libero lacus feugiat urna, eget hendrerit pede magna vitae lorem. Praesent mauris.

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit.

http://localhost/tahafuzalsunnah/live.php
One person recommends this.

http://localhost:8080/MyFBFriends/index.jsp
One person recommends this.

http://localhost/wordpress/?p=198
2 people recommend this.

http://localhost/?page_id=16
2 people recommend this.

http://localhost:9010/
2 people recommend this.

http://localhost/facebooktest/default.aspx
3 people recommend this.

Be the first to share your thoughts

Getting ready

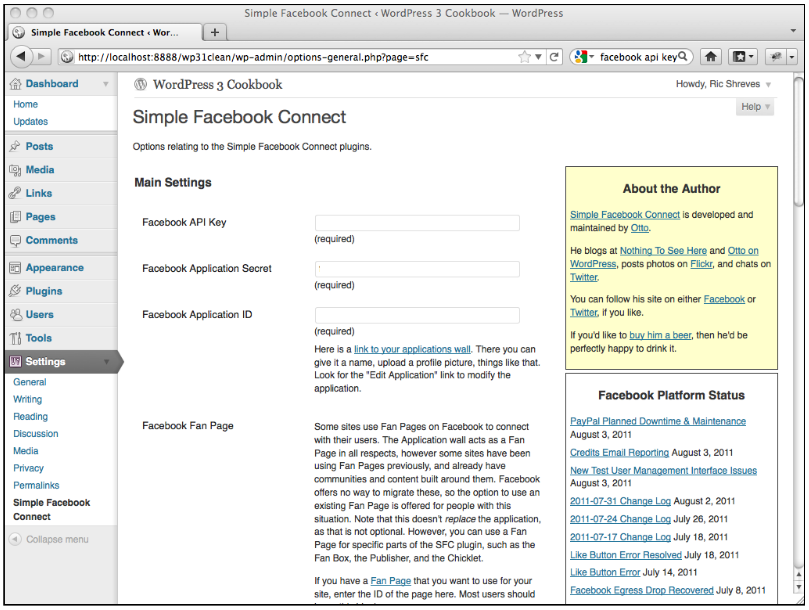
To execute this recipe, you will need to install the Simple Facebook Connect plugin. You will need to install this plugin before you can get started. Search for **Simple Facebook Connect** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it. Visit the installed plugins page to see the full list of items added by this plugin – it is extensive. Activate all those you wish to use.

> You can learn more about the plugin by visiting the developer's site at `http://ottopress.com/wordpress-plugins/simple-facebook-connect/`

Additionally you will need to have access to the Facebook API. To get the access you need, go to `https://developers.facebook.com/apps`, click on the **Create new App** button, and follow the steps outlined there.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on **Settings** on the main admin nav.
3. Click on the option **Simple Facebook Connect** in the **Settings** menu.
4. In the space provided, enter your **Facebook API Key**, as shown in the following screenshot:


Simple Facebook Connect ‹ WordPress 3 Cookbook — WordPress
Simple Facebook Connect ‹ Wor...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=sfc
facebook api key
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Simple Facebook Connect
Collapse menu
WordPress 3 Cookbook
Howdy, Ric Shreves
Help

Simple Facebook Connect

Options relating to the Simple Facebook Connect plugins.

Main Settings

Facebook API Key (required)

Facebook Application Secret (required)

Facebook Application ID (required)

Here is a link to your applications wall. There you can give it a name, upload a profile picture, things like that. Look for the "Edit Application" link to modify the application.

Facebook Fan Page

Some sites use Fan Pages on Facebook to connect with their users. The Application wall acts as a Fan Page in all respects, however some sites have been using Fan Pages previously, and already have communities and content built around them. Facebook offers no way to migrate these, so the option to use an existing Fan Page is offered for people with this situation. Note that this doesn't *replace* the application, as that is not optional. However, you can use a Fan Page for specific parts of the SFC plugin, such as the Fan Box, the Publisher, and the Chicklet.

If you have a Fan Page that you want to use for your site, enter the ID of the page here. Most users should

About the Author

Simple Facebook Connect is developed and maintained by Otto.

He blogs at Nothing To See Here and Otto on WordPress, posts photos on Flickr, and chats on Twitter.

You can follow his site on either Facebook or Twitter, if you like.

If you'd like to buy him a beer, then he'd be perfectly happy to drink it.

Facebook Platform Status

PayPal Planned Downtime & Maintenance August 3, 2011
Credits Email Reporting August 3, 2011
New Test User Management Interface Issues August 3, 2011
2011-07-31 Change Log August 2, 2011
2011-07-24 Change Log July 26, 2011
2011-07-17 Change Log July 18, 2011
Like Button Error Resolved July 18, 2011
Like Button Error July 14, 2011
Facebook Egress Drop Recovered July 8, 2011

5. Enter your **Facebook Application Secret**.
6. Enter your **Facebook Application ID**.
7. Click the **Save Changes** button.

Next, you need to get the widgets assigned to the pages.

1. Click on the **Appearance** menu.
2. Click on the **Widgets** option in the **Appearance** menu.
3. Click and drag all the widgets you wish to appear on the site.
4. Configure any widgets that require additional information.

How it works...

The functionality supplied by this plugin is all derived directly from Facebook. The plugin uses your APIR credentials to access the Facebook system and display the output on your pages.

> Expect that as Facebook makes changes to their offerings, that this plugin, and possibly the functionality, will change.

See also...

- *Adding social bookmarking buttons to your site*

Integrating a Twitter stream into your site

Twitter is a free micro blogging service that allows you to post short messages (less than 140 characters). Given the popularity of Twitter, we thought to include a recipe that allows you to integrate Twitter streams into your WordPress site.

There are quite a few plugins out there that let you bring Twitter content into your site. The Twitter Goodies Widget profiled in this recipe was chosen due to its flexibility and the fact that it relies on functionality provided directly by Twitter. The plugin allows you to display all four of the widget styles provided by Twitter and you can show these either in your site's widgets, or in your content items through the use of a shortcode.

> You can also grab the widget code directly from Twitter and integrate it into your site manually, if you prefer. Visit `http://twitter.com/about/resources/widgets`

Getting ready

To execute this recipe, you will need to install the Twitter Goodies Widget plugin. You will need to install this plugin before you can get started. Search for **Twitter Goodies Widget** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's site at `http://netweblogic.com/wordpress/plugins/twitter-goodies-widgets/`

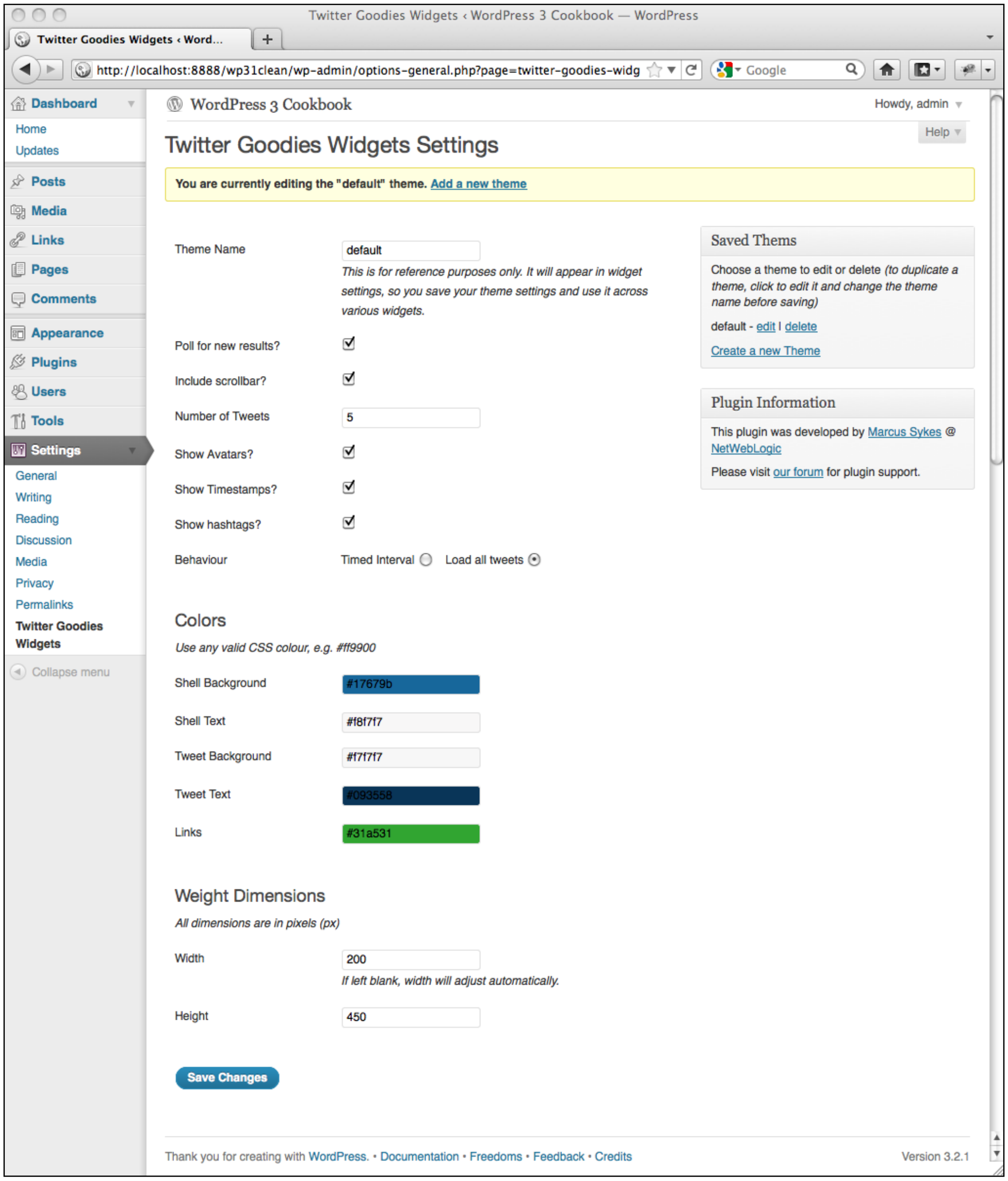
How to do it...

There are two parts to this recipe. First, you need to configure the plugin. Second, you need to set up the WordPress widget to display the output.

To configure the plugin, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the **Twitter Goodies Widget** link in the **Settings** menu.
4. Enter a name in the field marked **Theme Name**, as seen in the next screenshot. This name will appear in the widget options when you set up the widget later.
5. Select any other options you want on this page, including the color scheme.

6. Click the **Save Changes** button.


Twitter Goodies Widgets ‹ WordPress 3 Cookbook — WordPress
Twitter Goodies Widgets ‹ Word...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=twitter-goodies-widg
Google
Dashboard
Home
Updates
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Twitter Goodies Widgets
Collapse menu
WordPress 3 Cookbook
Howdy, admin
Help
Save Changes
Thank you for creating with WordPress. • Documentation • Freedoms • Feedback • Credits
Version 3.2.1

Twitter Goodies Widgets Settings

You are currently editing the "default" theme. Add a new theme

Theme Name: default

This is for reference purposes only. It will appear in widget settings, so you save your theme settings and use it across various widgets.

Poll for new results? ☑

Include scrollbar? ☑

Number of Tweets: 5

Show Avatars? ☑

Show Timestamps? ☑

Show hashtags? ☑

Behaviour: Timed Interval ○ Load all tweets ●

Colors

Use any valid CSS colour, e.g. #ff9900

Shell Background: #17679b

Shell Text: #f8f7f7

Tweet Background: #f7f7f7

Tweet Text: #093558

Links: #31a531

Weight Dimensions

All dimensions are in pixels (px)

Width: 200

If left blank, width will adjust automatically.

Height: 450

Saved Thems

Choose a theme to edit or delete *(to duplicate a theme, click to edit it and change the theme name before saving)*

default - edit | delete

Create a new Theme

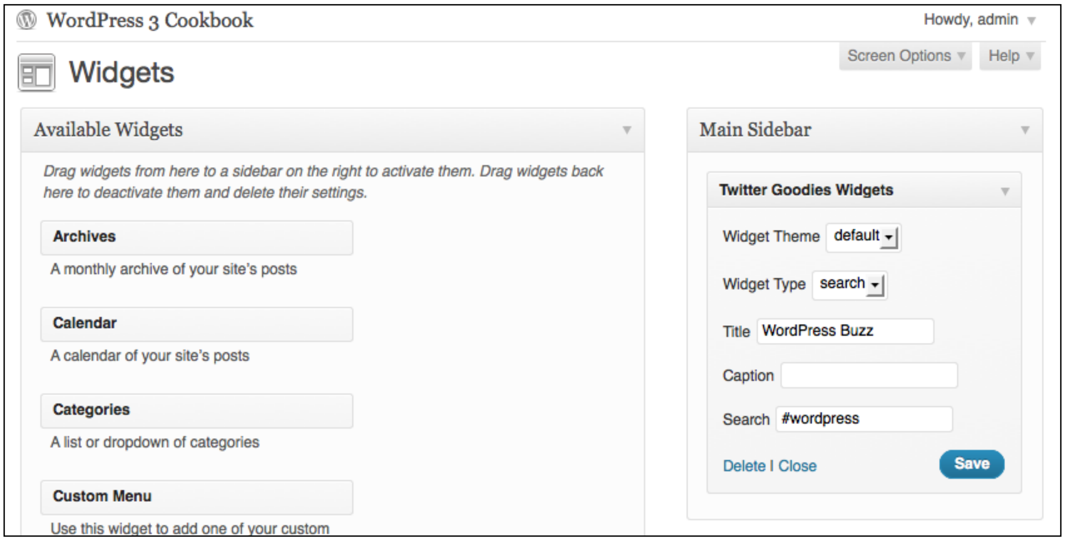
Plugin Information

This plugin was developed by Marcus Sykes @ NetWebLogic

Please visit our forum for plugin support.

The plugin is now ready to go. Let's place the output on the screen through the use of the WordPress widget that comes with the plugin. Follow these steps:

1. Click on the **Appearance** menu option.
2. Click on the **Widgets** choice on the **Appearance** menu.
3. Click and drag the widget named **Twitter Goodies Widget** to the sidebar where you want it to appear on your site.
4. Inside **the Twitter Goodies Widget**, select from the combo box the **Widget Theme** you wish to use (this is the theme you set up in the previous part of this recipe).
5. Select the **Widget Type** you want to display.
6. Most other values are optional, with some exceptions: If you select **List**, you will need to also choose the list to display. If you choose **Profile** or **Faves**, you must enter a username. If you choose **Search**, you must enter a search query.
7. When you've finished, click on the **Save** button.


WordPress 3 Cookbook
Howdy, admin
Screen Options
Help
Widgets
Available Widgets
Main Sidebar
Twitter Goodies Widgets
Delete | Close
Save

Drag widgets from here to a sidebar on the right to activate them. Drag widgets back here to deactivate them and delete their settings.

Archives
A monthly archive of your site's posts

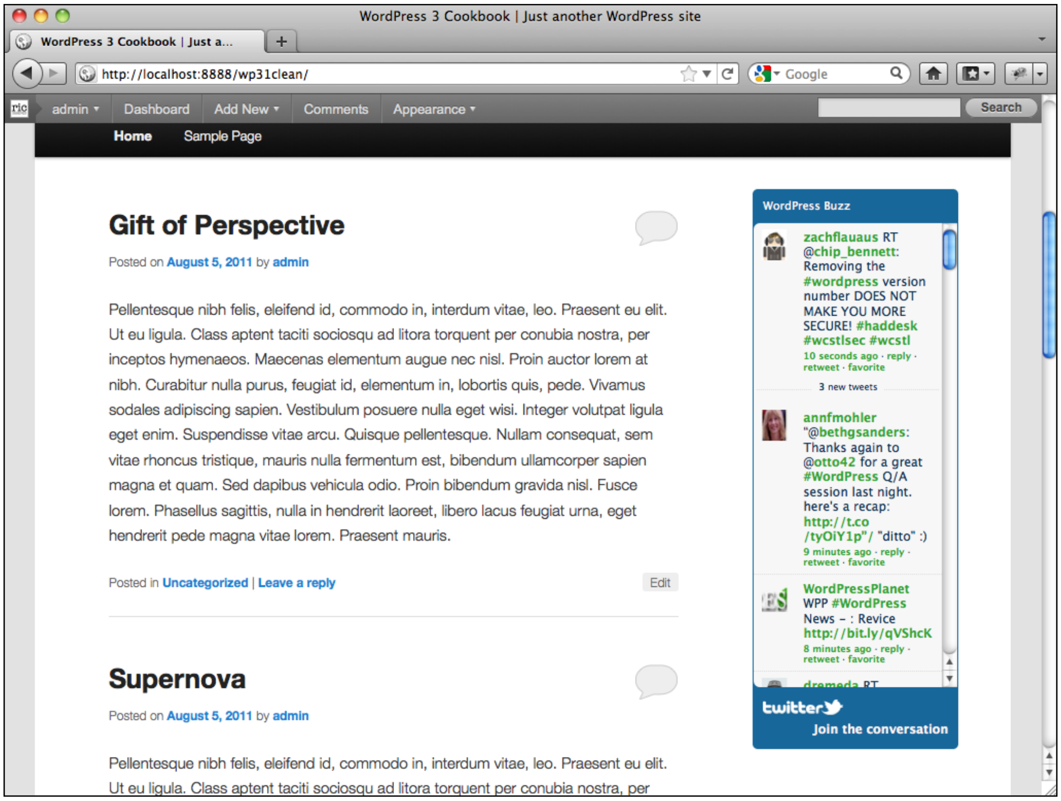
Calendar
A calendar of your site's posts

Categories
A list or dropdown of categories

Custom Menu
Use this widget to add one of your custom

Widget Theme: default
Widget Type: search
Title: WordPress Buzz
Caption:
Search: #wordpress

View the front end of your site to view the widget. In the next screenshot, you can see the Twitter search widget in action.



Gift of Perspective

Posted on August 5, 2011 by admin

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Maecenas elementum augue nec nisl. Proin auctor lorem at nibh. Curabitur nulla purus, feugiat id, elementum in, lobortis quis, pede. Vivamus sodales adipiscing sapien. Vestibulum posuere nulla eget wisi. Integer volutpat ligula eget enim. Suspendisse vitae arcu. Quisque pellentesque. Nullam consequat, sem vitae rhoncus tristique, mauris nulla fermentum est, bibendum ullamcorper sapien magna et quam. Sed dapibus vehicula odio. Proin bibendum gravida nisl. Fusce lorem. Phasellus sagittis, nulla in hendrerit laoreet, libero lacus feugiat urna, eget hendrerit pede magna vitae lorem. Praesent mauris.

Posted in Uncategorized | Leave a reply Edit

Supernova

Posted on August 5, 2011 by admin

Pellentesque nibh felis, eleifend id, commodo in, interdum vitae, leo. Praesent eu elit. Ut eu ligula. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per

zachflauaus RT @chip_bennett: Removing the #wordpress version number DOES NOT MAKE YOU MORE SECURE! #haddesk #wcstlsec #wcstl
10 seconds ago · reply · retweet · favorite

annfmohler "@bethgsanders: Thanks again to @otto42 for a great #WordPress Q/A session last night. here's a recap: http://t.co/tyOiY1p"/ "ditto" :)
9 minutes ago · reply · retweet · favorite

WordPressPlanet WPP #WordPress News – : Revice http://bit.ly/qVShcK
8 minutes ago · reply · retweet · favorite

dremeda RT

How it works...

The Twitter Goodies Widget leverages the widget capabilities offered by Twitter. Your choices mirror those options available on the Twitter site:

- **Faves**: Show all those tweets starred as favorites by a specific Twitter user account
- **List**: Show the members of one of the lists on a Twitter user's account
- **Profile**: Show the tweets from one specific Twitter user account
- **Search**: Set up a search of all Twitter users' posts

The various configuration options likewise mirror the choices offered by Twitter.

There's more...

As mentioned at the outset of this recipe, the plugin also supports inclusion of a Twitter stream inside of content items, through the use of a shortcode. To embed a Twitter stream, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Open for editing the content item in which you wish to display the Twitter stream.
3. Click on the **HTML** tab to switch to the HTML editor.
4. Type `[tgw]` where you want the Twitter Goodies Widget to appear.
5. Save the content item.

6

Implementing Online Sales and Advertising

In this chapter, we will cover:

- Integrating Google Adsense
- Displaying ads in your posts by using WordPress shortcodes
- Managing ad visibility
- Inserting ads into your RSS feeds
- Showing your site stats to find advertisers
- Enhancing your advertise page by adding Paypal subscriptions
- Managing your advertising space with an ad manager
- Adding a shopping cart to your site

Introduction

Though not all site owners are concerned with generating revenue from their site, if you are concerned with making money with your site, then WordPress is a good choice. Not only is WordPress flexible enough to support a variety of business models, there exists a large number of plugins that are designed to help you turn your site into a source of income.

Monetizing a site is typically accomplished through one of two methods: either selling advertising, or selling a product or service. In this chapter, we look at tools and techniques that can help you with either business model. The chapter includes plugins that let you insert and manage ads on your site and we also look at how to add a shopping cart to your site to support checkout of products or services.

One of the most popular advertising programs, Google AdSense, is discussed in several recipes in this chapter. Additionally, if you wish to manage your own ad space and inventory, later recipes cover ad management.

> While accepting payment online is an important topic, we focus only on PayPal in this chapter, due to the many variables users must overcome to establish a relationship with a payment gateway. Between regional variations, variations in site content, and other facts, it is not possible to set out steps and processes that will be applicable to everyone.

Integrating Adsense

Google AdSense is one of the most popular platforms for integrating the display of paid advertising into your site. The service, provided free of charge by Google, allows you to display ads on the pages of your website through the insertion of a simple code. Google manages the ads and keeps them relevant to your content. When a visitor clicks on an ad, a portion of the revenues Google earns is credited to you. Google will then periodically pay out the earnings directly to you.

While there are other, and more lucrative advertising channels, there are few that have the ease of use of Google AdSense.

In this recipe we look at how to integrate AdSense ads into your site.

> In the later recipes in this chapter we look at setting up and managing direct advertising on your site.

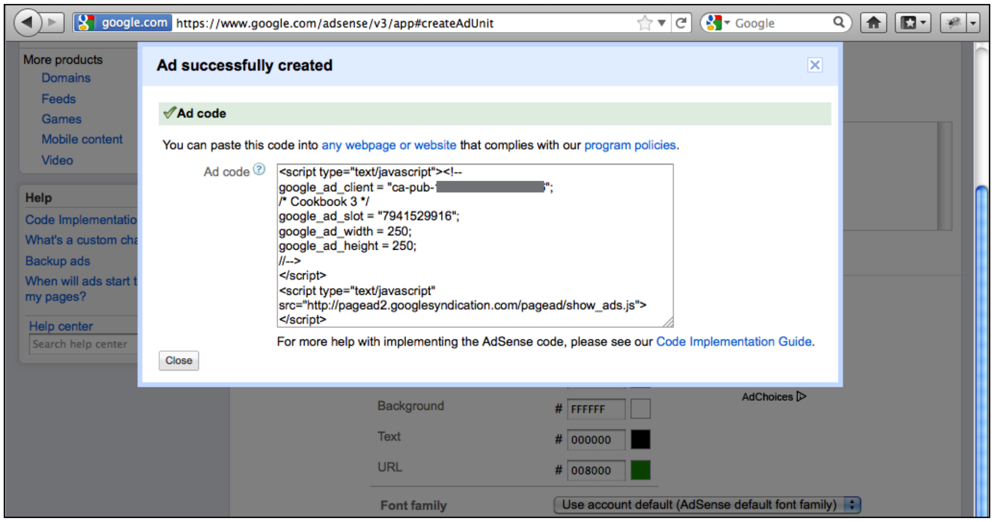
Getting ready

The first thing to do is to get the AdSense code to display ads on your blog. To get the code, connect to `https://www.google.com/adsense`. If you do not currently have an AdSense account, create one. If you already have a Google account for any service, you can add AdSense to that account. If you don't have a Google account, creating one is necessary to take part in the AdSense program.

Once you're connected to Google AdSense you have to create an advertisement. To do so, click on the **My ads** tab and click on the button labeled **New ad unit**.

You will be prompted to name and configure your new ad. One of the key steps is choosing an ad size. According to Google, the best performing ad sizes are 336 x 280, 300 x 250, and 160 x 600, but the decision ultimately depends on the space in your blog layout. Select the configuration options you prefer, and then click on the button labeled **Save and get code**.

As shown in the following screenshot, the next page will display your AdSense code. Copy that code; you will need it for this recipe.


google.com
https://www.google.com/adsense/v3/app#createAdUnit
Google
More products
Domains
Feeds
Games
Mobile content
Video
Help
Code Implementatio
What's a custom cha
Backup ads
When will ads start t
my pages?
Help center
Search help center
Ad code
Close
AdChoices
Font family
Use account default (AdSense default font family)

Ad successfully created

Ad code

You can paste this code into any webpage or website that complies with our program policies.

Ad code

```
<script type="text/javascript"><!--
google_ad_client = "ca-pub-[redacted]";
/* Cookbook 3 */
google_ad_slot = "7941529916";
google_ad_width = 250;
google_ad_height = 250;
//-->
</script>
<script type="text/javascript"
src="http://pagead2.googlesyndication.com/pagead/show_ads.js">
</script>
```

For more help with implementing the AdSense code, please see our Code Implementation Guide.

Background # FFFFFF

Text # 000000

URL # 008000

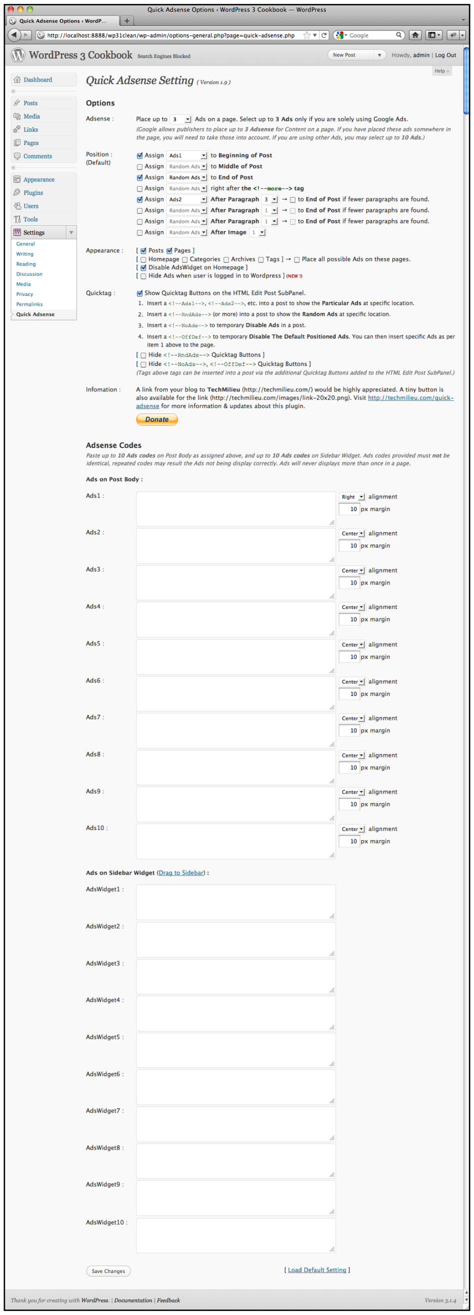
To execute this recipe, you will also need to install the Quick AdSense plugin. You will need to install this plugin before you can get started. Search for **Quick Adsense** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://quicksense.net/2009/09/16/quick-sense/`

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the option **Quick AdSense**.
4. From the **Position** controls, set where you want AdSense ads to appear on your site.
5. On the **Appearance** controls, decide whether you want the ads available on page, or posts, or a combination of these options.

6. Scroll down to the **AdSense Codes** section, as you can see in the following screenshot. Paste your AdSense code into the fields you need. For example, if you set **Ads1** to appear in the **Position** controls, then add your code into the field marked **Ads1**.


Quick Adsense Options ‹ WordPress 3 Cookbook — WordPress
Quick Adsense Options ‹ WordP...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=quick-adsense.php
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Quick Adsense

Quick Adsense Setting *(Version 1.9)*

Options

Adsense : Place up to 3 Ads on a page. Select up to **3 Ads** only if you are solely using Google Ads.
*(Google allows publishers to place up to 3 **Adsense** for Content on a page. If you have placed these ads somewhere in the page, you will need to take those into account. If you are using other Ads, you may select up to **10 Ads**.)*

Position : (Default)
☑ Assign Ads1 to **Beginning of Post**
☐ Assign Random Ads to **Middle of Post**
☑ Assign Random Ads to **End of Post**
☐ Assign Random Ads right after **the <!--more--> tag**
☑ Assign Ads2 **After Paragraph** 3 → ☐ to **End of Post** if fewer paragraphs are found.
☐ Assign Random Ads **After Paragraph** 1 → ☐ to **End of Post** if fewer paragraphs are found.
☐ Assign Random Ads **After Paragraph** 1 → ☐ to **End of Post** if fewer paragraphs are found.
☐ Assign Random Ads **After Image** 1

Appearance :
[☑ Posts ☑ Pages]
[☐ Homepage ☐ Categories ☐ Archives ☐ Tags] → ☐ Place all possible Ads on these pages.
[☑ Disable AdsWidget on Homepage]
[☐ Hide Ads when user is logged in to Wordpress] (NEW !)

Quicktag : ☑ Show Quicktag Buttons on the HTML Edit Post SubPanel.

1. Insert a `<!--Ads1-->`, `<!--Ads2-->`, etc. into a post to show the **Particular Ads** at specific location.
2. Insert a `<!--RndAds-->` (or more) into a post to show the **Random Ads** at specific location.
3. Insert a `<!--NoAds-->` to temporary **Disable Ads** in a post.
4. Insert a `<!--OffDef-->` to temporary **Disable The Default Positioned Ads**. You can then insert specific Ads as per item 1 above to the page.

[☐ Hide `<!--RndAds-->` Quicktag Buttons]
[☐ Hide `<!--NoAds-->`, `<!--OffDef-->` Quicktag Buttons]
(Tags above tags can be inserted into a post via the additional Quicktag Buttons added to the HTML Edit Post SubPanel.)

Infomation : A link from your blog to **TechMilieu** (http://techmilieu.com/) would be highly appreciated. A tiny button is also available for the link (http://techmilieu.com/images/link-20x20.png). Visit http://techmilieu.com/quick-adsense for more information & updates about this plugin.

Adsense Codes

*Paste up to **10 Ads codes** on Post Body as assigned above, and up to **10 Ads codes** on Sidebar Widget. Ads codes provided must **not** be identical, repeated codes may result the Ads not being display correctly. Ads will never displays more than once in a page.*

Ads on Post Body :

Ads1 : Right alignment, 10 px margin
Ads2 : Center alignment, 10 px margin
Ads3 : Center alignment, 10 px margin
Ads4 : Center alignment, 10 px margin
Ads5 : Center alignment, 10 px margin
Ads6 : Center alignment, 10 px margin
Ads7 : Center alignment, 10 px margin
Ads8 : Center alignment, 10 px margin
Ads9 : Center alignment, 10 px margin
Ads10 : Center alignment, 10 px margin

Ads on Sidebar Widget (Drag to Sidebar) **:**

AdsWidget1 :
AdsWidget2 :
AdsWidget3 :
AdsWidget4 :
AdsWidget5 :
AdsWidget6 :
AdsWidget7 :
AdsWidget8 :
AdsWidget9 :
AdsWidget10 :

Save Changes
[Load Default Setting]
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.4

7. If you want to use widget positions for your ads, insert the code into the fields under the heading **Ads on Sidebar Widget**.
8. When you've finished, click **Save Changes**.

Your ads will now appear automatically, as per your settings. If you wish to display the ads in the widget areas, you will, however, also have to access the Widgets Manager and drag the appropriate ad widgets into the widget areas you wish to use.

How it works...

Once you insert your code into the plugin, your work is done. The plugin will handle the placement on the screen and Google will manage the ads. In the next section of this recipe we look how some other options for using AdSense, including manual placement of the code to bypass the plugin.

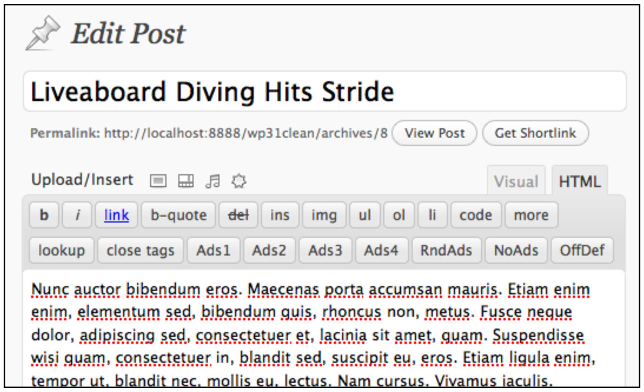
There's more...

The Quick AdSense plugin offers you another interesting option, that is, adding ads directly into posts or pages where you want them. To use this feature of the plugin, you will use what the plugin calls quicktags – a concept very similar to WordPress shortcodes.

Integrating AdSense using quicktags

Once the plugin is activated, you also have to make sure the Quicktags options are selected in the plugin configuration. This is enabled by default, as you can see in the preceding screenshot. Once you have confirmed that the feature is enabled, follow these steps:

1. Open for editing the page or post to which you wish to add the AdSense ads.
2. Click the **HTML** tab to switch to the HTML editor.
3. You will note on the editor bar there is a set of new buttons, corresponding to the number of ads you set up. You can see the buttons in the following screenshot:



Liveaboard Diving Hits Stride

Permalink: http://localhost:8888/wp31clean/archives/8

Nunc auctor bibendum eros. Maecenas porta accumsan mauris. Etiam enim enim, elementum sed, bibendum quis, rhoncus non, metus. Fusce neque dolor, adipiscing sed, consectetuer et, lacinia sit amet, quam. Suspendisse wisi quam, consectetuer in, blandit sed, suscipit eu, eros. Etiam ligula enim, tempor ut, blandit nec, mollis eu, lectus. Nam cursus. Vivamus iaculis.

4. Place the cursor where you want the ad to appear.
5. Click on the button for the ad number you want to appear.
6. Click on the **Update** button to finish and update the post.

Integrating AdSense manually

The first method is to insert the code directly into your theme files. To do so, follow these simple steps:

1. Open the file where you'd like the ads to be displayed, for example, `index.php` (your homepage), `sidebar.php` (your blog sidebar), or `single.php` (single posts pages).
2. Paste your AdSense code. Make sure that you're not inserting your Google AdSense code within PHP tags,(`<?php` and `?>`) as you'll probably get a parsing error.
3. Save the file. Your AdSense ads are now displayed on your blog. Please note that sometimes it can take up to 15 minutes until the first ads are displayed.

The second method is to use a text widget to display your AdSense ads.

Text widgets can't display PHP code, but they can display the JavaScript code used by Google AdSense.

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.
3. Click on the **Widgets** option.
4. Add a text widget to your sidebar.
5. Paste your AdSense code in the widget body.
6. You can also give a title to the widget, but this isn't recommended. Google AdSense terms and conditions don't allow text such as **Please visit** or **Recommended** above AdSense ads. You shouldn't give any title to the widget.
7. Click on the **Save** button of the widget to record your changes in this widget.

Displaying ads anywhere in your posts by using WordPress shortcodes

In the previous recipe, we looked at how to add Google AdSense ads to your site by using a plugin or by using a text widget. However, there is another option—enabling your site to display ads anywhere in your posts through the use of a WordPress shortcode.

Introduced in WordPress 2.5, shortcodes are powerful WordPress functions.

> If you use forums such as phpBB or Vbulletin, you're probably familiar with the use of shortcodes; the concept in WordPress is similar.

One of the most powerful features of WordPress shortcodes is that you are able to create your own shortcodes. In this recipe, you'll learn to create a WordPress shortcode dedicated to inserting AdSense ads anywhere in your posts.

Getting ready

To execute this recipe, you'll need an AdSense code, as described in the previous recipe. You will also need a code editor and need access to your WordPress theme files on your server.

How to do it...

Follow these simple steps to get started:

1. Open your `functions.php` file.
2. Enter the following code. Don't forget to replace the AdSense code provided here by yours!

```
function cookbook_showads() {
     return '<script type="text/javascript"><!--
     google_ad_client = "ca-pub-xxxxxxxxxxxx";
   google_ad_slot = "7941529916";google_ad_width = 250;
     google_ad_height = 250;//-->
</script>
<script type="text/javascript"
  src="http://pagead2.googlesyndication.com/pagead/show_ads.js">
</script>
';
}
add_shortcode('adsense', 'cookbook_showads');
```

3. Save your `functions.php` file.

Your system now has a new shortcode available: `adsense`. To insert it into a post, follow these additional steps:

1. Open any of your published posts for editing.
2. Switch to the HTML editor by clicking on the **HTML** tab.

3. Insert the following shortcode anywhere in the post code, where you'd like AdSense to be displayed:

   ```
   [adsense]
   ```

4. Save your post.

How it works...

Shortcodes are handled by a set of functions called the `shortcode API`. When a post is displayed, post content is parsed and the shortcode API automatically transforms shortcodes into what they're supposed to be.

> You can learn more by visiting the WordPress API at `http://codex.wordpress.org/Shortcode_API`

In this recipe, we started by creating a very simple PHP function, called `cookbook_showads()`, to return the AdSense code. Then, we used the `add_shortcode()` function to create a shortcode by using the `cookbook_showads()` function.

Managing ad visibility

All web site owners who are interested in selling advertising on their site need to be concerned about **click through rates** (**CTRs**). For example, your CTR is one percent if your ads are displayed to 100 visitors and only one of them clicked on it.

A high CTR is usually a sign of good matching of ads to content and to visitors. The higher your CTR is, the more money Google earns from your ads. If they display some ads on your website, but no one clicks on them, they don't earn any money and that space is a waste for them.

Low CTR is also an issue for you as a site owner. Many AdSense users who have a CTR below one or two percent see their AdSense earnings dramatically decrease. Google calls this policy Smart Pricing, and as a site owner, it means that clicks on your ads will only get about 10 percent of what they are normally worth. As we can't do anything to fight the Smart Pricing policy, the best thing you can do is to try and avoid being Smart Priced.

Many search engine specialists agree that people coming from search engines tend to click more on the ads on a site. Your readership (such as RSS subscribers) don't often click on your ads.

In this recipe, we'll look at a plugin that helps you display your ads to the most likely visitors, while hiding them from those less likely to click. The net effect is that you will see an improvement in your CTR. The plugin is called Who Sees Ads.

Getting ready

To execute this recipe, you will need to install the Who Sees Ads plugin. You will need to install this plugin before you can get started. Search for **Who Sees Ads** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

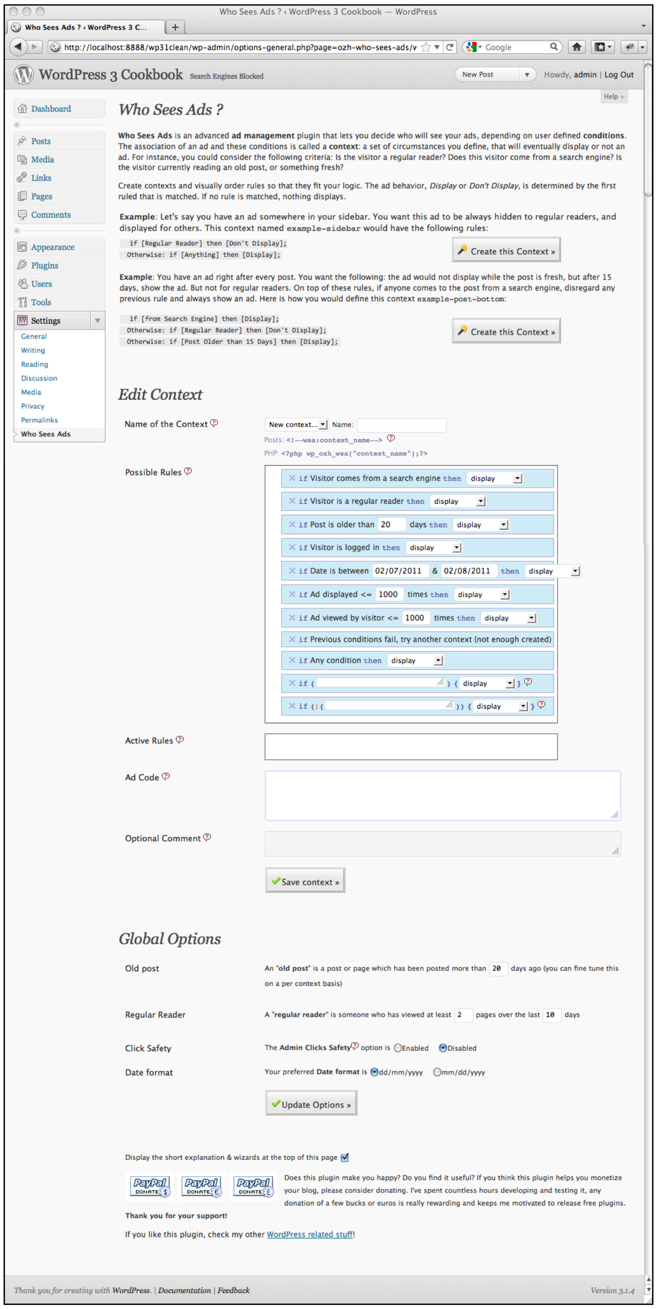
> You can learn more about the plugin by visiting the developer's website at `http://planetozh.com/blog/my-projects/wordpress-plugin-who-sees-ads-control-adsense-display/`

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the option **Who Sees Ads**.
4. On the **Who Sees Ads** configuration screen, shown in the following screenshot, the first thing you have to do is to create a context. A context is simply a condition, or a set of conditions, that control the display of an ad. By default, you can select one or more of the following:
    - If visitor comes from a search engine, display / don't display
    - If visitor is a regular reader, display / don't display
    - If post is older than XX days, display / don't display
    - If visitor is logged in, display / don't display
    - If date is between specified date interval, display / don't display
    - If ad has been showed fewer than XX times, display / don't display
    - If this visitor has viewed this ad fewer than XX times, display / don't display
    - If all previous conditions fail, try another context
    - If any condition, display / don't display
5. Provide a name for the context in the **Name of the Context** field.
6. Select one or more of the possible rules.
7. In the **Ad code** text area, paste your AdSense (or other) code.
8. Click on the **Save context** to save your context. It is now ready for use.

9. To insert ads, use the following code in your posts (after turning WordPress editor to HTML mode):

```
<!--wsa:context_name-->
```


Who Sees Ads ? ‹ WordPress 3 Cookbook — WordPress
Who Sees Ads ? ‹ WordPress 3 C...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=ozh-who-sees-ads/v
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Who Sees Ads

Who Sees Ads ?

Who Sees Ads is an advanced **ad management** plugin that lets you decide who will see your ads, depending on user defined **conditions**. The association of an ad and these conditions is called a **context**: a set of circumstances you define, that will eventually display or not an ad. For instance, you could consider the following criteria: Is the visitor a regular reader? Does this visitor come from a search engine? Is the visitor currently reading an old post, or something fresh?

Create contexts and visually order rules so that they fit your logic. The ad behavior, *Display* or *Don't Display*, is determined by the first ruled that is matched. If no rule is matched, nothing displays.

Example: Let's say you have an ad somewhere in your sidebar. You want this ad to be always hidden to regular readers, and displayed for others. This context named `example-sidebar` would have the following rules:

```
 if [Regular Reader] then [Don't Display];
Otherwise: if [Anything] then [Display];
```

Create this Context »

Example: You have an ad right after every post. You want the following: the ad would not display while the post is fresh, but after 15 days, show the ad. But not for regular readers. On top of these rules, if anyone comes to the post from a search engine, disregard any previous rule and always show an ad. Here is how you would define this context `example-post-bottom`:

```
 if [from Search Engine] then [Display];
Otherwise: if [Regular Reader] then [Don't Display];
Otherwise: if [Post Older than 15 Days] then [Display];
```

Create this Context »

Edit Context

Name of the Context: New context... Name:

Posts: `<!--wsa:context_name-->`

PHP: `<?php wp_ozh_wsa("context_name");?>`

Possible Rules

- if Visitor comes from a search engine then display
- if Visitor is a regular reader then display
- if Post is older than 20 days then display
- if Visitor is logged in then display
- if Date is between 02/07/2011 & 02/08/2011 then display
- if Ad displayed <= 1000 times then display
- if Ad viewed by visitor <= 1000 times then display
- if Previous conditions fail, try another context (not enough created)
- if Any condition then display
- if ( ) { display }
- if (!( )) { display }

Active Rules

Ad Code

Optional Comment

Save context »

Global Options

Old post: An **"old post"** is a post or page which has been posted more than 20 days ago (you can fine tune this on a per context basis)

Regular Reader: A **"regular reader"** is someone who has viewed at least 2 pages over the last 10 days

Click Safety: The **Admin Clicks Safety** option is ○ Enabled ● Disabled

Date format: Your preferred **Date format** is ● dd/mm/yyyy ○ mm/dd/yyyy

Update Options »

Display the short explanation & wizards at the top of this page ☑

PayPal DONATE
PayPal DONATE
PayPal DONATE

Does this plugin make you happy? Do you find it useful? If you think this plugin helps you monetize your blog, please consider donating. I've spent countless hours developing and testing it, any donation of a few bucks or euros is really rewarding and keeps me motivated to release free plugins.

Thank you for your support!

If you like this plugin, check my other WordPress related stuff!

Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.4

How it works...

The **Who Sees Ads** plugin is a complex plugin. When you create a context, it is recorded in the WordPress database. When a visitor comes to the site, a PHP conditional statement is created and appended to your theme files and posts. This code contains PHP conditional statements to make sure that the current visitor matches the conditions established in the context.

There's more...

Like any other very advanced plugin, **Who Sees Ads** has a lot of options. Now that you have learned how to install and configure this powerful plugin, let's look at the available configuration options.

Global options

The **Who Sees Ads** plugin also allows you to set up global options such as what defines a regular reader, what is an old post, and so on.

Here are the possible options:

- **Old post**: The number of days after which a post is considered old
- **Regular reader**: How many pages a visitor must have seen in the last X days to be considered as a regular reader
- **Click safety**: If enabled, this option will not display any AdSense or Yahoo ads to the blog administrator
- **Date format**: You can choose to use two different date formats—the American mm/dd/yyyy or the European dd/mm/yyyy

Once you're done with the global options, simply click on the **Update Options** button to save it.

Inserting ads into your RSS feeds

Your WordPress site comes with RSS feeds ready to go. Over time, users will subscribe to your feed to receive notifications of new content on your site. A portion of those RSS subscribers may rarely visit your site, preferring instead to simply use their feed reader to read your articles. If you want to reach those users with a promotional message, you need a way to embed that message into the RSS feed. If you're using Feedburner and Google AdSense, the solution is simple, as Feedburner allows you to insert AdSense ads into your feeds. However, if you want to insert other ads or promotional messages into your feed, you will need another solution.

In this recipe we look at a plugin that allows us to insert additional information into your site's feed.

Getting ready

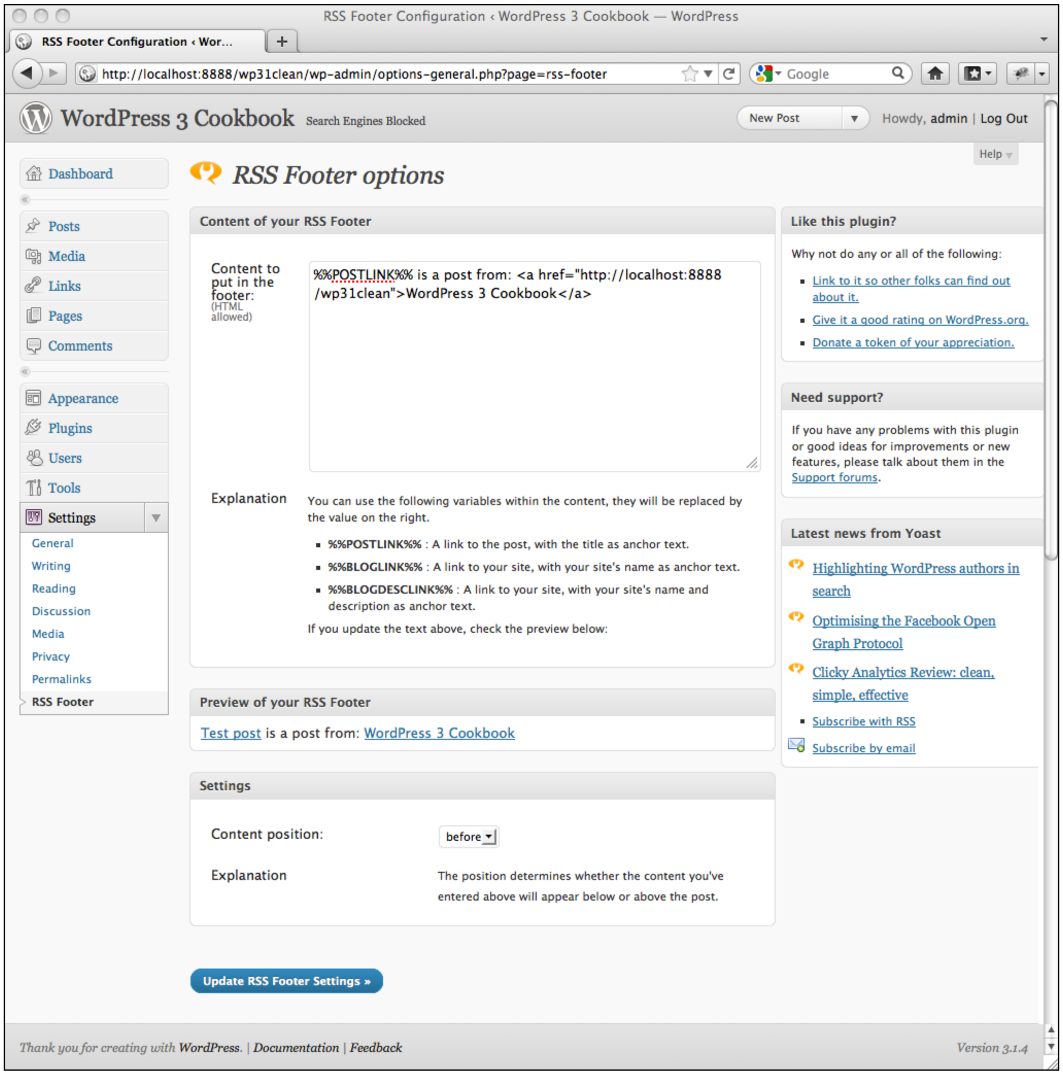
To execute this recipe, you will need to install the **RSS Footer** plugin. You will need to install this plugin before you can get started. Search for **Rss Footer** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://yoast.com/wordpress/rss-footer/`

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the option **RSS Footer**.
4. On the **RSS Footer options** page, seen in the following screenshot, you can input the content you want to add to the RSS feed, and position it on the screen. Type your message in the text box at the top of the page.
5. From the **Settings** combo box, select whether the message will appear above or below the feed items.

6. Click on the **Update RSS Footer Settings** button.



RSS Footer Configuration ‹ WordPress 3 Cookbook — WordPress
RSS Footer Configuration ‹ Wor...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=rss-footer
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
RSS Footer

RSS Footer options

Content of your RSS Footer

Content to put in the footer: (HTML allowed)

%%POSTLINK%% is a post from: WordPress 3 Cookbook

Explanation

You can use the following variables within the content, they will be replaced by the value on the right.

- %%POSTLINK%% : A link to the post, with the title as anchor text.
- %%BLOGLINK%% : A link to your site, with your site's name as anchor text.
- %%BLOGDESCLINK%% : A link to your site, with your site's name and description as anchor text.

If you update the text above, check the preview below:

Preview of your RSS Footer

Test post is a post from: WordPress 3 Cookbook

Settings

Content position: before

Explanation

The position determines whether the content you've entered above will appear below or above the post.

Update RSS Footer Settings »

Like this plugin?

Why not do any or all of the following:

- Link to it so other folks can find out about it.
- Give it a good rating on WordPress.org.
- Donate a token of your appreciation.

Need support?

If you have any problems with this plugin or good ideas for improvements or new features, please talk about them in the Support forums.

Latest news from Yoast

- Highlighting WordPress authors in search
- Optimising the Facebook Open Graph Protocol
- Clicky Analytics Review: clean, simple, effective
- Subscribe with RSS
- Subscribe by email

Thank you for creating with WordPress. | Documentation | Feedback

Version 3.1.4

How it works...

The plugin simply adds new information to the RSS feed output.

There's more

If you are using Feedburner to manage the RSS feed for your site, you may have to take further steps to make sure the new footer message is distributed with your feed items.

First things first – after implementing the recipe, visit your Feedburner feed and see whether your RSS Footer message is appearing. If it is, then you are fine and no further steps are needed. If the content is not appearing, you will need to follow these additional steps:

1. Log in to your Feedburner account.
2. Click on the name of your site's feed.
3. Look for the **Troubleshootize** tab at the top of the page. Click on it.
4. On the page that loads, scroll down to find the button **Resynch Now**. Click on that button.

If you were successful, you should see a confirmation message in your browser. You can then check again your feed output to see if the RSS Footer message appears.

Showing your site stats to find advertisers

In this chapter, we have talked a lot about Google AdSense as a means of monetizing your website. However, another effective way to make money of your site is to directly sell the advertising space on your site.

Undertaking direct ad sales has implications for the site owner: You need to market the site, process payments, and provide an advertising management system. In this recipe, we look at the first issue and discuss how to help promote your site through the display of performance statistics.

Displaying select site performance statistics can help attract advertisers and also help shape advertiser expectations and justify pricing. The data you show should help convince the prospective advertisers that your site is an appropriate place for them to promote their products or services.

In terms of advancing the cause of advertising sales, you should display a mix of statistics that show how popular your site is, how well it ranks, and how much reach it has. As this sort of data changes over time, you don't want to hardcode it; rather you want these statistics to be current and to automatically update themselves.

This recipe covers implementing the WPStatsDashboard plugin as a means of enhancing site promotion. Display this widget on a page containing your advertising rates and information to create a page that will help prospective advertisers connect with your site.

Getting ready

The WP Stats Dashboard relies on the statistics plugin from WordPress.com. In *Chapter 1*, we looked at installing JetPack, a plugin from Automattic that incorporates the WordPress Stats functionality. Before you set up WP Stats Dashboard, you will need to have the Jetpack plugin installed and properly configured. You will also need to know your Wordpress.com username and password.

To execute this recipe, you will need to install the **WP Stats Dashboard** plugin. You will need to install this plugin before you can get started. Search for **WP Stats Dashboard** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://www.daveligthart.com/wp-stats-dashboard-10/`

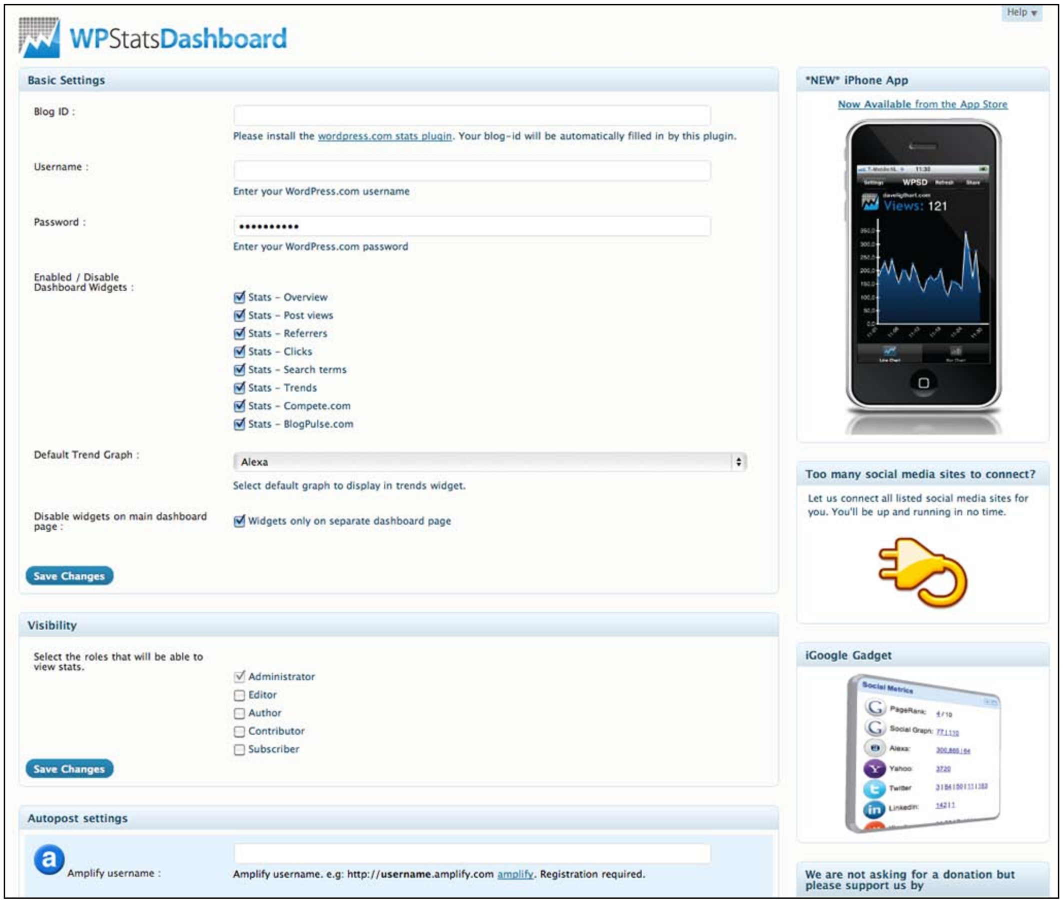
How to do it...

There are two parts to this recipe. In the first part, we will configure the plugin to produce for us the stats we want to display. In the second part, we will set up and display a widget containing these statistics.

Let's start by configuring the plugin:

1. Log in to your WordPress **Dashboard**.
2. Click on the **WPSD** menu.

3. Click on the option **Settings**. The following screenshot shows part of this very long screen:



Basic Settings

Blog ID :

Please install the wordpress.com stats plugin. Your blog-id will be automatically filled in by this plugin.

Username :

Enter your WordPress.com username

Password : ••••••••••

Enter your WordPress.com password

Enabled / Disable Dashboard Widgets :

☑ Stats – Overview
☑ Stats – Post views
☑ Stats – Referrers
☑ Stats – Clicks
☑ Stats – Search terms
☑ Stats – Trends
☑ Stats – Compete.com
☑ Stats – BlogPulse.com

Default Trend Graph : Alexa

Select default graph to display in trends widget.

Disable widgets on main dashboard page : ☑ Widgets only on separate dashboard page

Visibility

Select the roles that will be able to view stats.

☑ Administrator
☐ Editor
☐ Author
☐ Contributor
☐ Subscriber

Autopost settings

Amplify username :

Amplify username. e.g: http://**username**.amplify.com amplify. Registration required.

Too many social media sites to connect?

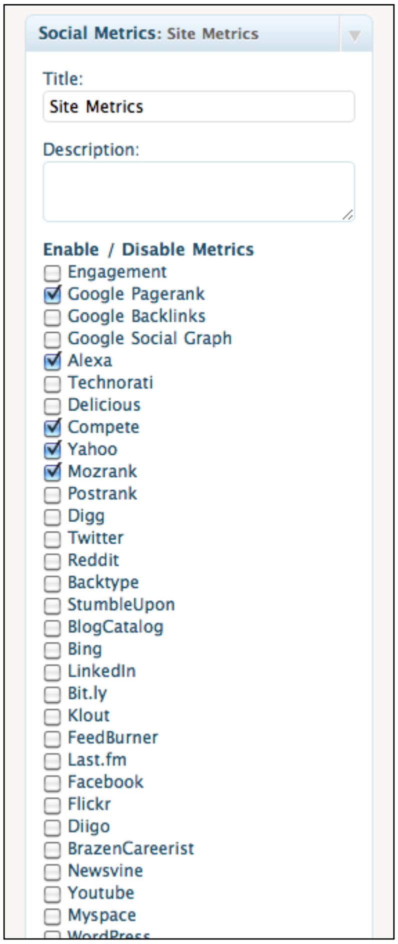
Let us connect all listed social media sites for you. You'll be up and running in no time.

We are not asking for a donation but please support us by

4. Make sure the **Blog ID** field, seen at the top of the page in the preceding image, is filled in. Normally this will happen automatically if you have properly installed and configured the WordPress Stats or Jetpack plugin.
5. Add your WordPress.com username, in the field marked **Username**.
6. Add your WordPress.com password, in the field marked **Password**.
7. Click on **Save changes**.
8. Scroll down the page to the **Profile Settings** section of the page. You can complete any of the options you want here, but for advertising sales purposes, you will probably want to make sure to complete the fields for any chapters where your site performance is particularly strong.
9. Click on the **Save Changes** button at the bottom of the **Profile Settings** section.

The plugin is now ready for your use. The next part of the recipe is to configure and position a widget to display select information for your advertising page. To do that, follow these steps:

1. Click on the **Appearance** menu.
2. Click on the option **Widgets**.
3. Find the **Social Metrics** widget and drag it to the widget area where you wish it to appear.
4. Click on it to open the widget for editing.
5. Give it a name, for example, **Site Statistics.**
6. Click on it to select from the list the statistics you feel show your site in the best light, as shown in the following image:



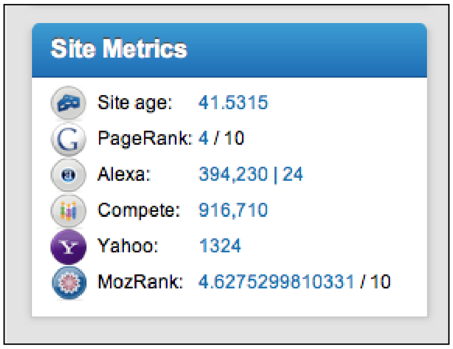
Title: Site Metrics

Description:

Enable / Disable Metrics
☐ Engagement
☑ Google Pagerank
☐ Google Backlinks
☐ Google Social Graph
☑ Alexa
☐ Technorati
☐ Delicious
☑ Compete
☑ Yahoo
☑ Mozrank
☐ Postrank
☐ Digg
☐ Twitter
☐ Reddit
☐ Backtype
☐ StumbleUpon
☐ BlogCatalog
☐ Bing
☐ LinkedIn
☐ Bit.ly
☐ Klout
☐ FeedBurner
☐ Last.fm
☐ Facebook
☐ Flickr
☐ Diigo
☐ BrazenCareerist
☐ Newsvine
☐ Youtube
☐ Myspace
☐ WordPress

7. Click on the **Save** button.

Your site statistics are now visible on your site, as seen in the following screenshot:



Site Metrics

Site age:	41.5315
PageRank:	4 / 10
Alexa:	394,230 | 24
Compete:	916,710
Yahoo:	1324
MozRank:	4.6275299810331 / 10

For best effect, publish the widget to the page containing your pricing information and contact details.

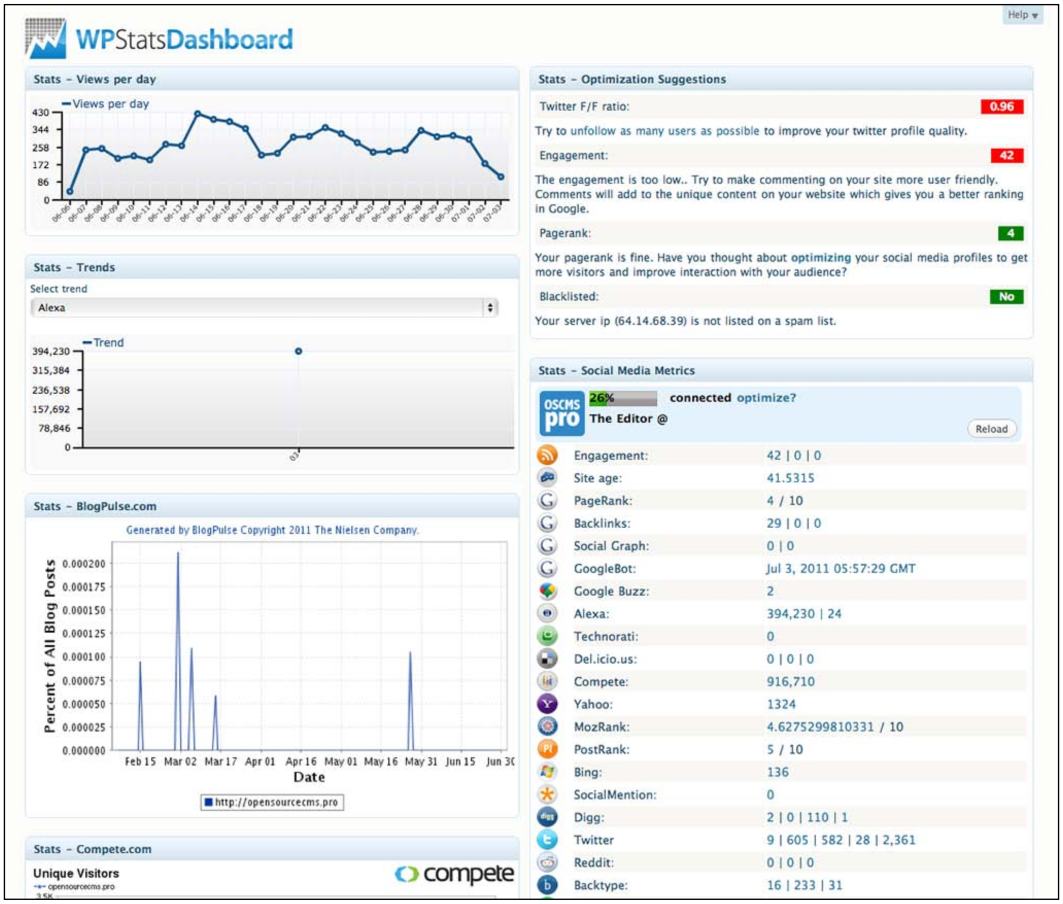
How it works...

This plugin gathers data from multiple sources. It interfaces with the WordPress.com statistics system to pull in site traffic stats, but then also looks to third party sites for additional information, from traffic and site rank, to social bookmarking activity, to intangibles such as Klout.

There's more

This plugin also supplies are very thorough statistics dashboard for your site. You will want to explore the options for displaying admin dashboard widgets. On the WPSD configuration screen, shown on the first screenshot in this section, there are multiple options that relate to the display of the statistics inside your WordPress dashboard. To get the most out of the plugin, select the dashboard widgets you want from the list labeled **Enable / Disable Dashboard Widgets**. You will also want to select the option marked **Disable widgets on main dashboard page** in order to avoid performance problems on your main dashboard page.

Once you have made your selections, save your changes, then click on the option **Dashboard Stats**, under the **Dashboard** menu. When you click on that link, you will be taken to a dashboard similar to the one shown in the following screenshot:


Help ▾
WPStatsDashboard
Stats – Views per day
Views per day
430
344
258
172
86
0
06-06
06-07
06-08
06-09
06-10
06-11
06-12
06-13
06-14
06-15
06-16
06-17
06-18
06-19
06-20
06-21
06-22
06-23
06-24
06-25
06-26
06-27
06-28
06-29
06-30
07-01
07-02
07-03
Stats – Trends
Select trend
Alexa
Trend
394,230
315,384
236,538
157,692
78,846
0
03
Stats – BlogPulse.com
Generated by BlogPulse Copyright 2011 The Nielsen Company.
Percent of All Blog Posts
0.000200
0.000175
0.000150
0.000125
0.000100
0.000075
0.000050
0.000025
0.000000
Feb 15
Mar 02
Mar 17
Apr 01
Apr 16
May 01
May 16
May 31
Jun 15
Jun 30
Date
http://opensourcecms.pro
Stats – Compete.com
Unique Visitors
opensourcecms.pro
compete
Stats – Optimization Suggestions
Stats – Social Media Metrics
OSCMS pro
26%
Reload

Twitter F/F ratio: 0.96

Try to unfollow as many users as possible to improve your twitter profile quality.

Engagement: 42

The engagement is too low.. Try to make commenting on your site more user friendly. Comments will add to the unique content on your website which gives you a better ranking in Google.

Pagerank: 4

Your pagerank is fine. Have you thought about optimizing your social media profiles to get more visitors and improve interaction with your audience?

Blacklisted: No

Your server ip (64.14.68.39) is not listed on a spam list.

connected optimize?

The Editor @

Engagement:	42 | 0 | 0
Site age:	41.5315
PageRank:	4 / 10
Backlinks:	29 | 0 | 0
Social Graph:	0 | 0
GoogleBot:	Jul 3, 2011 05:57:29 GMT
Google Buzz:	2
Alexa:	394,230 | 24
Technorati:	0
Del.icio.us:	0 | 0 | 0
Compete:	916,710
Yahoo:	1324
MozRank:	4.6275299810331 / 10
PostRank:	5 / 10
Bing:	136
SocialMention:	0
Digg:	2 | 0 | 110 | 1
Twitter	9 | 605 | 582 | 28 | 2,361
Reddit:	0 | 0 | 0
Backtype:	16 | 233 | 31

Enhancing your Advertise page by adding Paypal subscriptions

If you have applied the previous recipe to your blog, you now have a stunning **Advertise** page with your site stats. That's great, but how can the advertisers pay you? You could use a contact form and ask advertisers to e-mail you and then send them your Paypal ID, but that process is slow and loaded with opportunities for the buyer to change their mind before completing the payment. A better solution would be to provide a payment option directly from your Advertise page. Even better would be setting up recurring payments so that the buyers can subscribe to your advertising service on an ongoing basis.

On this recipe we look at setting up Paypal to enable recurring or subscription payment for your site.

Getting ready

To allow Paypal subscriptions on your WordPress blog, you need a Paypal account. If you don't already have a Paypal account, just go to `www.paypal.com` and create one.

How to do it...

Once you have your Paypal account ready, follow these steps to create a subscription for your ads.

1. First, log in to Paypal.
2. Click on **Merchant Services**.
3. In the **Merchant Tools** page, click on the **Subscriptions & Recurring Payments** button.
4. Complete the form and specify your subscription options.
5. You need to fill in the following details:
    - **Accept payments for**: Leave the default, Subscriptions and recurring billing.
    - **Item name**: Give a name to the subscription, as shown in the preceding example.
    - **Subscription ID**: You can leave this field blank.
    - **Customize button**: Only useful if you wish to allow the advertiser to define the ads own price.

- **Currency**: The currency you'd like to display your prices in. You should choose USD, as US dollars are a de facto standard on the Internet, except if your blog is in any other language than English.
- **Recurring amount to be billed**: The sum you'd like to get each month.
- **Billing cycle**: Choose the time period for the subscription. On most blogs, 1 month is the default.
- **After how many cycles should billing stop**: You should leave the field as is, however you can decide that the subscription should be canceled automatically after X cycles.

6. Once done, click on the **Create** button. The next page will provide your custom subscription code.
7. Copy the code.
8. Log in to your WordPress **Dashboard**.
9. Open for editing the page where you have your advertising information for site visitors.
10. Paste the code where you want the payment button to appear.

Your advertisers can now automatically pay for their ads on a monthly basis.

Managing your advertising space with an ad manager

If you wish to display multiple ads on your site, or you wish to use an ad network other than Google AdSense, you may want to explore the use of an ad manager for your website. Ad management systems typically include the range of options needed to maximize the use of the advertising space on your site. The ad manager is also the key in situations where you wish to run campaigns by time, impressions, or clicks.

There are a number of options for ad management, from relatively simple systems to complete software packages you have to install independently on your server. This recipe looks at the Simple Ads Manager plugin for WordPress. It is a relatively complex plugin, but offers a fair amount of power in exchange. If you wish to sell time-based, or click-based ads, or if you wish to divide your site's pages into different zones for displaying ads, then Simple Ads Manager is a good choice for you.

Getting ready

To execute this recipe, you will need to install the **Simple Ads Manager** plugin. You will need to install this plugin before you can get started. Search for "Simple ads manager" inside the **Add New Plugins** screen of your WordPress site. After you find it, click to install, and then activate it.

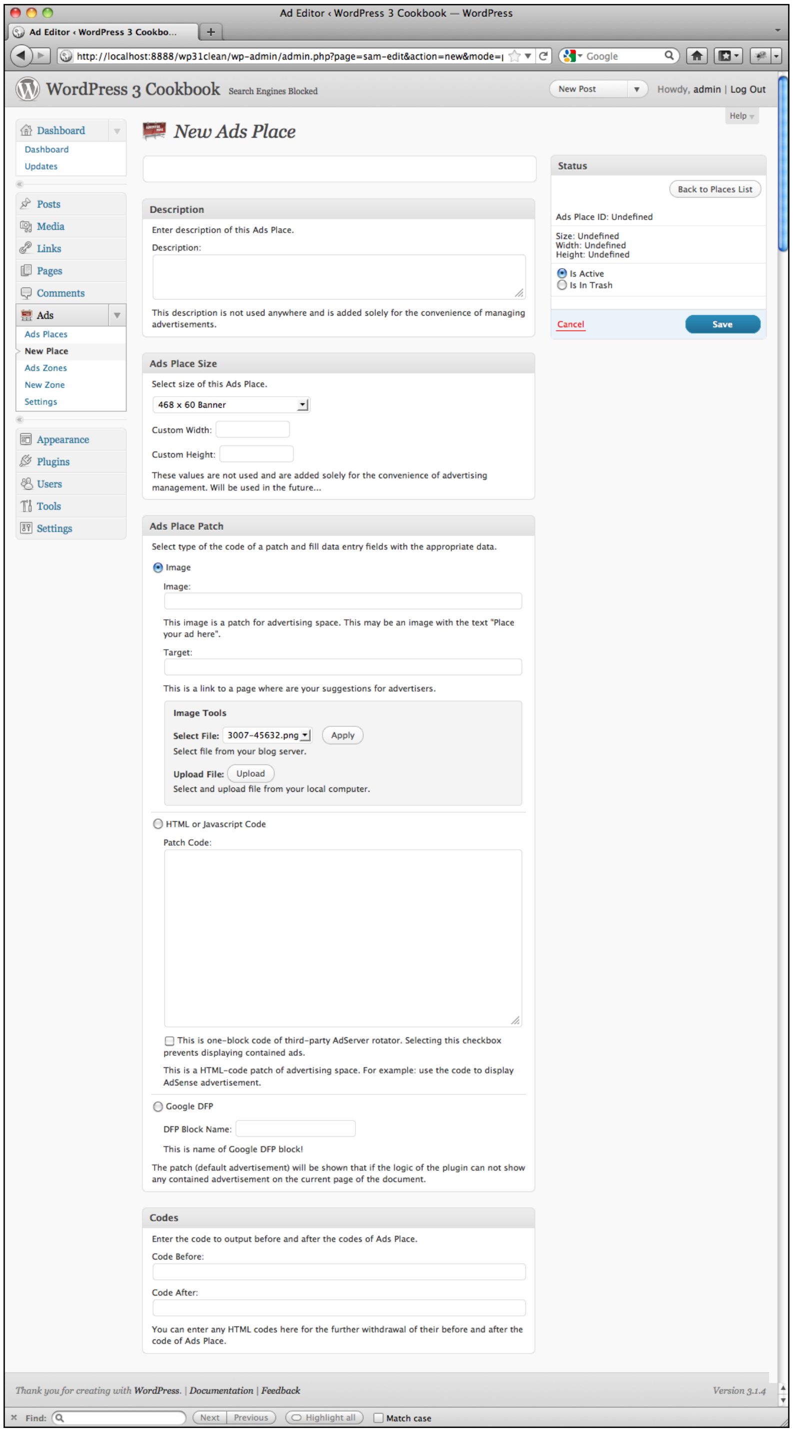
> You can learn more about the plugin by visiting the developer's website at `http://www.simplelib.com/?p=480`

In addition to the plugin, you will need the ads you want to display, or if you plan to use AdSense, the code.

How to do it...

There are several steps to this recipe. The Simple Ads Manager groups all ads inside of what the plugin calls Ad Places. The first task we need to complete is the creation of at least one Ad Place. To create your site's first Ad Place, follow these steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Ads** menu.
3. Click on the option **New Place**, or click on the button marked **Add New Place**.
4. On the **New Ads Place** screen that appears (see the following screenshot) enter a logical name for the **Ads Place** in the first field. You can also give it a **Description** if you wish.


Ad Editor ‹ WordPress 3 Cookbook — WordPress
Ad Editor ‹ WordPress 3 Cookbo...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=sam-edit&action=new&mode=
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Dashboard
Updates
Posts
Media
Links
Pages
Comments
Ads
Ads Places
New Place
Ads Zones
New Zone
Settings
Appearance
Plugins
Users
Tools
Settings
Back to Places List
Cancel
Save
Apply
Upload
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.4
Find:
Next
Previous
Highlight all
☐ Match case

New Ads Place

Status

Ads Place ID: Undefined

Size: Undefined
Width: Undefined
Height: Undefined

● Is Active
○ Is In Trash

Description

Enter description of this Ads Place.

Description:

This description is not used anywhere and is added solely for the convenience of managing advertisements.

Ads Place Size

Select size of this Ads Place.

468 x 60 Banner

Custom Width:

Custom Height:

These values are not used and are added solely for the convenience of advertising management. Will be used in the future...

Ads Place Patch

Select type of the code of a patch and fill data entry fields with the appropriate data.

● Image

Image:

This image is a patch for advertising space. This may be an image with the text "Place your ad here".

Target:

This is a link to a page where are your suggestions for advertisers.

Image Tools

Select File: 3007-45632.png

Select file from your blog server.

Upload File:

Select and upload file from your local computer.

○ HTML or Javascript Code

Patch Code:

☐ This is one-block code of third-party AdServer rotator. Selecting this checkbox prevents displaying contained ads.

This is a HTML-code patch of advertising space. For example: use the code to display AdSense advertisement.

○ Google DFP

DFP Block Name:

This is name of Google DFP block!

The patch (default advertisement) will be shown that if the logic of the plugin can not show any contained advertisement on the current page of the document.

Codes

Enter the code to output before and after the codes of Ads Place.

Code Before:

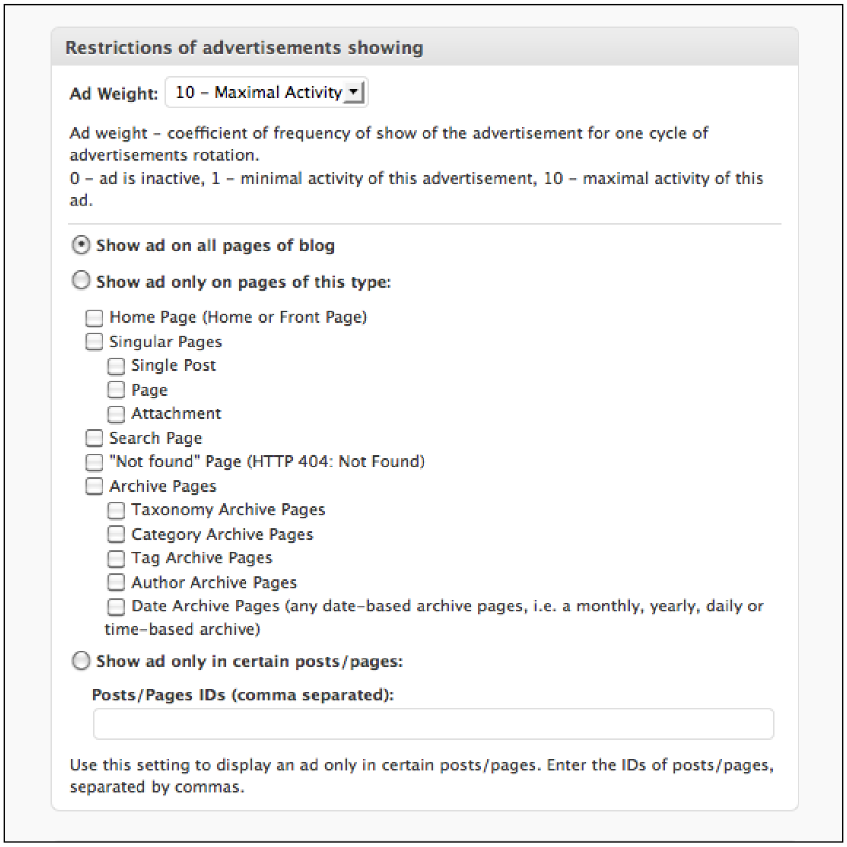
Code After:

You can enter any HTML codes here for the further withdrawal of their before and after the code of Ads Place.

5. Next, in the section marked **Ads Place Size**, select a size for your ad, or enter a custom width and height in the fields provided.
6. In the **Ads Place Patch**, select whether this place will hold image ads, HTML / javascript ads or ads, from Google DoubleClick for Publishers (Google DFP).
7. Click on the **Save** button.

Next, you need to create at least one ad to run in your Ad Place. To create an ad, follow these steps:

1. Click on the **Ads** menu.
2. Click on the option **Ads Place**.
3. On the **Managing Ad Places** page, hover the mouse over the name of the Ad Place you wish to hold your new ad. On the options that appear, click **New Ad**.
4. The **New Advertisement** screen is where you set up the actual ad. In the first field on the page, give your ad a name. Add a **Description** if you wish.
5. In the **Ad Code** section of the page, select whether you wish to use **Image Mode** (an image ad) or **Code Mode** (an HTML or javascript ad).
6. If you selected Image Mode, upload the image you want to use. If you selected Code Mode, paste the code in the field given.
7. Next, set the display conditions for the ad in the section titled **Restrictions of advertisements showing** (see image below). Note that the section immediately below, labeled **Extended restrictions**, allows you to make even more specific conditions on the display.
8. Add **Prices**, if relevant.
9. Click on the **Save** button.



Restrictions of advertisements showing

Ad Weight: 10 – Maximal Activity

Ad weight – coefficient of frequency of show of the advertisement for one cycle of advertisements rotation.
0 – ad is inactive, 1 – minimal activity of this advertisement, 10 – maximal activity of this ad.

● **Show ad on all pages of blog**

○ **Show ad only on pages of this type:**

- ☐ Home Page (Home or Front Page)
- ☐ Singular Pages
  - ☐ Single Post
  - ☐ Page
  - ☐ Attachment
- ☐ Search Page
- ☐ "Not found" Page (HTTP 404: Not Found)
- ☐ Archive Pages
  - ☐ Taxonomy Archive Pages
  - ☐ Category Archive Pages
  - ☐ Tag Archive Pages
  - ☐ Author Archive Pages
  - ☐ Date Archive Pages (any date-based archive pages, i.e. a monthly, yearly, daily or time-based archive)

○ **Show ad only in certain posts/pages:**

Posts/Pages IDs (comma separated):

Use this setting to display an ad only in certain posts/pages. Enter the IDs of posts/pages, separated by commas.

Finally, it's time to place the ads on the page. To display an ad, follow these steps:

1. Click on the **Appearance** menu.
2. Click on the **Widgets** option.
3. Grab the **Ads Place** widget and drag it to the widget area where you want the ad to appear.
4. Click on the **Ads Place** widget to open it for editing.
5. Select the name of the Ads Place you just created from the **Ads Place** combo box.
6. Click the **Save** button.

Your ad will now appear on your pages in the position you placed them.

How it works...

Ad Places are used to hold one or more individual ads. The Ad Places widget will allow you to then place the ads on the page. If you put more than one ad into an Ad Place, the ads can be set to rotate. If you need to create multiple ads in different positions, simply create multiple Ad Places.

There's more...

The plugin also provides support for Ads Zones. Zones are containers that can hold one or more Ad Places. Zones are most useful for grouping together content pages for the purpose of controlling ad display.

Creating a new Ad Zone

1. Log in to your WordPress **Dashboard**.
2. Click on the **Ads** menu.
3. Click on the **New Zone** option.
4. In the first field on the New Ads Zone page, shown in the next screenshot, give your new zone a name. Add a **description** if you wish.
5. In the **Ads Zone Settings** section, set up which Ad Places you wish to use. You can set a default Place to show where no specifics are provided, or you can vary the Places used by a variety of criteria.
6. Click on the **Save** button.
7. Click on the **Appearance** menu.
8. Click on the **Widgets** option.
9. Find the **Ads Zone** widget and drag it to the widget area where you want the ads to appear.
10. Click on the **Ads Zone** widget to open it for editing.
11. Select which Zone you want to use from the **Ads Zone** combo box.
12. Click on the **Save** button.

Adding a shopping cart to your site

If you want to sell products on your site, at the very least you will need a way for people to add those products to a shopping cart and go through a checkout process to pay for the items. In this recipe, we look at a basic e-commerce solution. The plugin covered in this recipe enables you to create an **Add to Cart** button for your products and also provides a shopping cart widget. Once the user decides to checkout, the plugin will send the users to Paypal for payment.

Getting ready

To execute this recipe, you will need to install the WP Paypal Shopping Cart plugin. You will need to install this plugin before you can get started. Search for **WP Paypal Shopping Cart** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://www.tipsandtricks-hq.com/wordpress-simple-paypal-shopping-cart-plugin-768`

How to do it...

There are three parts to this recipe. First, we need to configure the plugin. Second, we need to enable the Shopping Cart widget. Finally, we need to add the shortcode to our products.

Let's start by configuring the plugin; here's how to do it:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the option **WP Shopping Cart**.
4. On the page that loads, you will see a large number of configuration options. The next screenshot shows the **WP Paypal Shopping Cart Options** page. You will need to go through this page and select the options necessary for your shop, including your Paypal e-mail address, your preferred currency, whether shipping costs are involved, and so on. You can also configure the text that will appear for your cart and the **Buy now** button.
5. Once you have selected the options you wish, click the **Update Options** button.

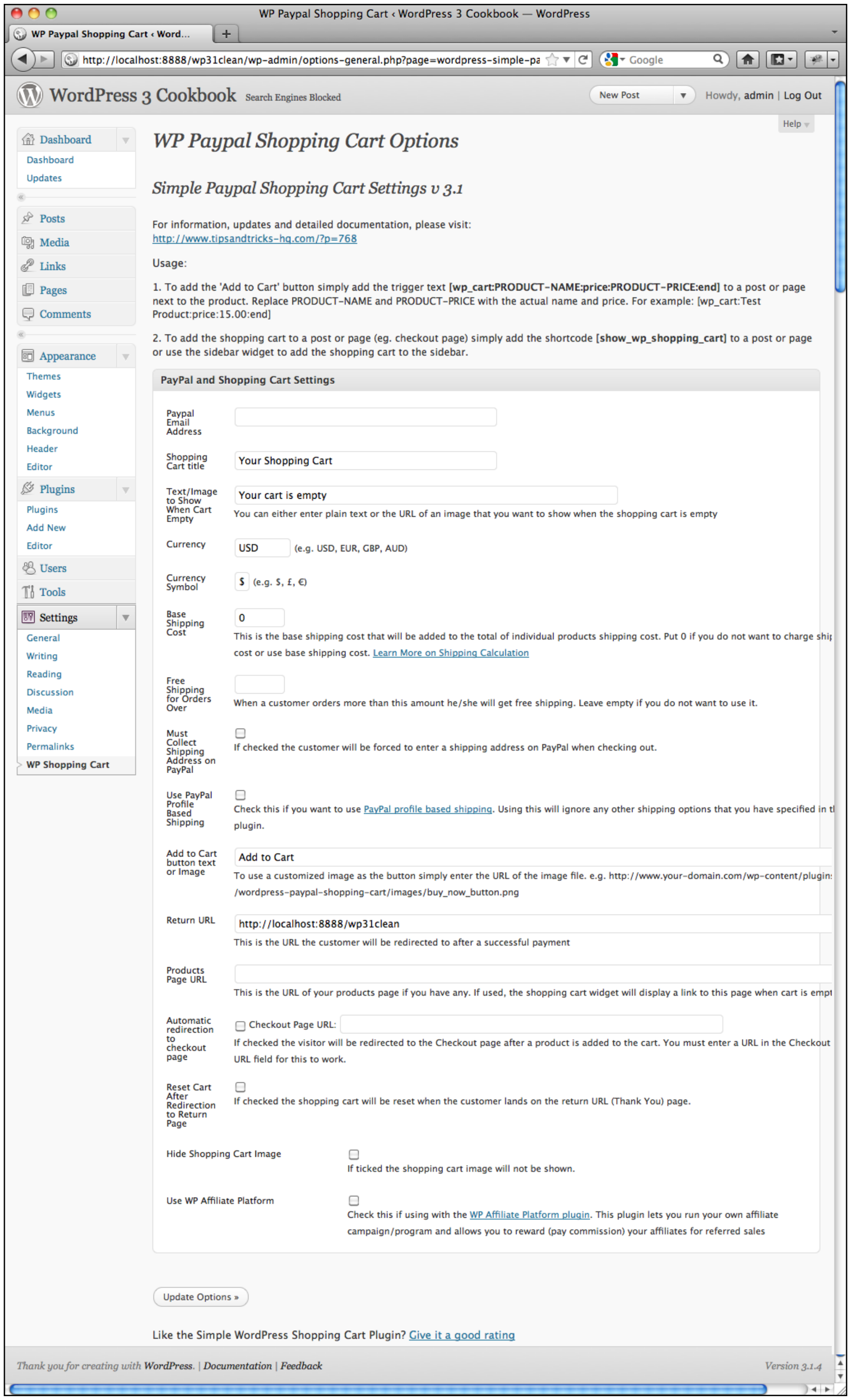
The system is now ready to go. Let's next place the shopping cart link on the page for your site visitors. Follow these steps:

1. Click on the **Appearance** menu.
2. Click on the **Widgets** option.
3. Find the **WP Paypal Shopping Cart** widget and drag it to the widget area where you want it to appear on your site.

The final step is to place the appropriate code on your product page to enable the purchase. Follow these steps:

1. Go to the page where you want the product with the checkout option to appear.
2. Click and open the page (or post) for editing.
3. Click on the **HTML** tab to switch to the HTML editor.
4. Where you want the **Add to Cart** button to appear, paste the following code into the text: `[wp_cart:PRODUCT-NAME:price:PRODUCT-PRICE:end]`. Replace **PRODUCT-NAME** with the name of your product and **PRODUCT-PRICE** with the price of the product, for example, `[wp_cart:WordPress Book:price:29.95:end]`.
5. Click to update the page.

That's all there is to it. You can use the plugin to create **Add to Cart** buttons for as many unique products as you like. All products will add to the shopping cart. Users will be able to remove things from the cart or add new items before checking out.


WP Paypal Shopping Cart ‹ WordPress 3 Cookbook — WordPress
WP Paypal Shopping Cart ‹ Word...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=wordpress-simple-pa
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Dashboard
Updates
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Background
Header
Editor
Plugins
Plugins
Add New
Editor
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
WP Shopping Cart

WP Paypal Shopping Cart Options

Simple Paypal Shopping Cart Settings v 3.1

For information, updates and detailed documentation, please visit:
http://www.tipsandtricks-hq.com/?p=768

Usage:

1. To add the 'Add to Cart' button simply add the trigger text **[wp_cart:PRODUCT-NAME:price:PRODUCT-PRICE:end]** to a post or page next to the product. Replace PRODUCT-NAME and PRODUCT-PRICE with the actual name and price. For example: [wp_cart:Test Product:price:15.00:end]

2. To add the shopping cart to a post or page (eg. checkout page) simply add the shortcode **[show_wp_shopping_cart]** to a post or page or use the sidebar widget to add the shopping cart to the sidebar.

PayPal and Shopping Cart Settings

Paypal Email Address:

Shopping Cart title: Your Shopping Cart

Text/Image to Show When Cart Empty: Your cart is empty
You can either enter plain text or the URL of an image that you want to show when the shopping cart is empty

Currency: USD (e.g. USD, EUR, GBP, AUD)

Currency Symbol: $ (e.g. $, £, €)

Base Shipping Cost: 0
This is the base shipping cost that will be added to the total of individual products shipping cost. Put 0 if you do not want to charge shi cost or use base shipping cost. Learn More on Shipping Calculation

Free Shipping for Orders Over:
When a customer orders more than this amount he/she will get free shipping. Leave empty if you do not want to use it.

Must Collect Shipping Address on PayPal: ☐
If checked the customer will be forced to enter a shipping address on PayPal when checking out.

Use PayPal Profile Based Shipping: ☐
Check this if you want to use PayPal profile based shipping. Using this will ignore any other shipping options that you have specified in t plugin.

Add to Cart button text or Image: Add to Cart
To use a customized image as the button simply enter the URL of the image file. e.g. http://www.your-domain.com/wp-content/plugin /wordpress-paypal-shopping-cart/images/buy_now_button.png

Return URL: http://localhost:8888/wp31clean
This is the URL the customer will be redirected to after a successful payment

Products Page URL:
This is the URL of your products page if you have any. If used, the shopping cart widget will display a link to this page when cart is empt

Automatic redirection to checkout page: ☐ Checkout Page URL:
If checked the visitor will be redirected to the Checkout page after a product is added to the cart. You must enter a URL in the Checkout URL field for this to work.

Reset Cart After Redirection to Return Page: ☐
If checked the shopping cart will be reset when the customer lands on the return URL (Thank You) page.

Hide Shopping Cart Image: ☐
If ticked the shopping cart image will not be shown.

Use WP Affiliate Platform: ☐
Check this if using with the WP Affiliate Platform plugin. This plugin lets you run your own affiliate campaign/program and allows you to reward (pay commission) your affiliates for referred sales

Update Options »

Like the Simple WordPress Shopping Cart Plugin? Give it a good rating

Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.4

How it works...

This plugin does not create a product catalog, nor does it manage the display of products; it simply enables an "add to cart" button and provides the users with a shopping cart interface that enables payment via Paypal. If you want to style your product display, you will need to lay it out inside a content item and style it to suit your needs.

When a plugin is installed, it adds a new shortcode functionality to your site. The configuration options you set are used to provide the basic information that is used both in the shopping cart display and in the transaction data to Paypal.

The shopping cart contains most of the critical user interfaces and interfaces with the Paypal system for getting the payment information to that system.

There's more...

The shopping cart is available both as a widget, and as a shortcode that can be added into a content item. If you prefer to use the shortcode to place the shopping cart, simply add the code `[show_wp_shopping_cart]` to your content item.

7

Making an SEO Friendly Site

In this chapter you will learn:

- Making your site visible to search engines
- Optimizing your permalinks for SEO
- Migrating your permalinks safely
- Adding redirects for changed URLs
- Creating meta descriptions for your posts and pages
- Avoiding duplicate content with a robots.txt file
- Pinging third party services
- Enhancing site indexing with XML sitemaps
- Using Google's and Bing's Webmaster tools
- Improving SEO with the SEO Ultimate plugin

Introduction

Building a great site is one thing, getting people to visit it is another thing entirely. From a marketing perspective, making your site search engine friendly is essential to your success in driving traffic to the site.

Search engine optimization, or SEO, as it is commonly known, is more of an art than a science. Though the experts are unlikely to agree to one specific approach that is key to your SEO success, there are a number of techniques that are widely recognized as effective in helping you to achieve your goals.

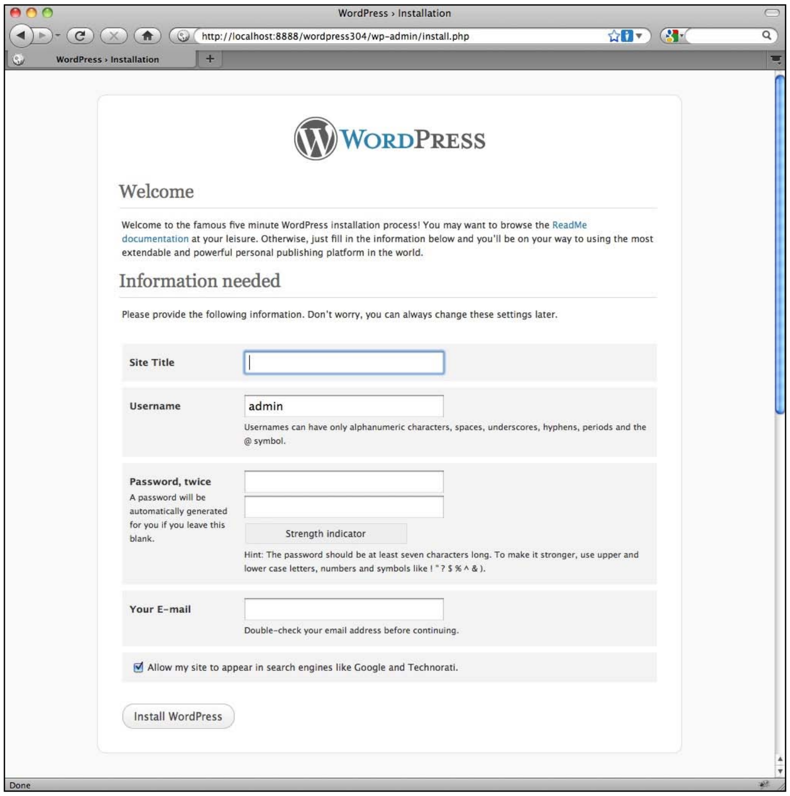
In this chapter, we look at some of the most effective tools and techniques for making your WordPress site more search engine friendly. Topics covered include how to create search engine friendly URLS, how to enhance your metadata, how to avoid duplicate content penalties, and how to register with third-party services that can help boost your search engine visibility.

Effective search engine marketing requires planning, coordination, and a dedication to consistent effort over time. There are no fast fixes. You should consider the recipes in this chapter as a good starting point, not a definitive final solution.

Making your site visible to search engines

You have to set your WordPress configuration to allow the search engines to index the site. If you do not allow the site to be visible to the search engines, then your site's pages will not be included in the search engine indexes and consequently, will not appear in the search results.

During the installation process, the WordPress installer prompts you to make a decision whether to make the site visible to the search engines. In the following screenshot you can see where this appears in the installer:



Welcome

Welcome to the famous five minute WordPress installation process! You may want to browse the ReadMe documentation at your leisure. Otherwise, just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Don't worry, you can always change these settings later.

Site Title:

Username: admin

Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods and the @ symbol.

Password, twice

A password will be automatically generated for you if you leave this blank.

Hint: The password should be at least seven characters long. To make it stronger, use upper and lower case letters, numbers and symbols like ! " ? $ % ^ &).

Your E-mail:

Double-check your email address before continuing.

☑ Allow my site to appear in search engines like Google and Technorati.

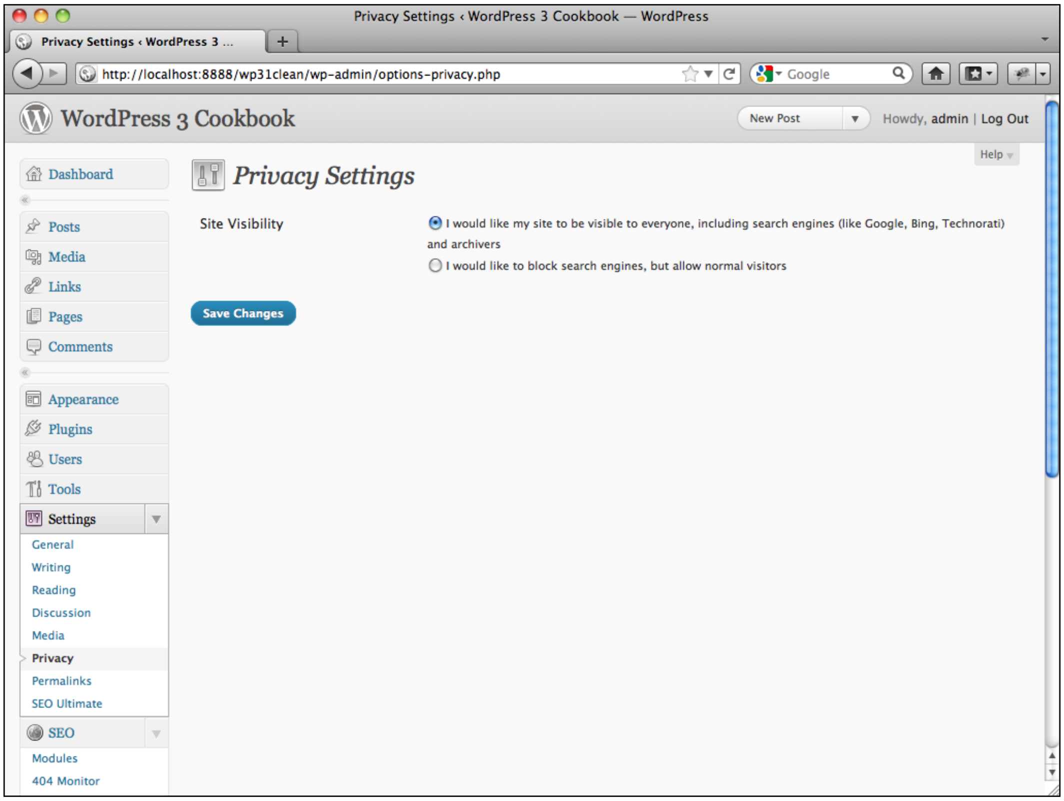
However, regardless of the setting you selected during installation, you can adjust this after the site is installed. In this recipe, we show you how to change this setting.

Getting ready

Nothing extra is needed to cook up this recipe; everything is done from within the WordPress dashboard.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Go to **Settings**.
3. Select the option **Privacy,** as shown in the following screenshot:



Privacy Settings ‹ WordPress 3 Cookbook — WordPress

http://localhost:8888/wp31clean/wp-admin/options-privacy.php

WordPress 3 Cookbook

Privacy Settings

Site Visibility

● I would like my site to be visible to everyone, including search engines (like Google, Bing, Technorati) and archivers

○ I would like to block search engines, but allow normal visitors

Privacy Settings ‹ WordPress 3 ...
Google
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
SEO Ultimate
SEO
Modules
404 Monitor
Save Changes

4. Select the option **I would like my site to be visible to everyone, including search engines (like Google, Bing, Technorati) and archivers.**
5. Click on **Save Changes.**

Optimizing your permalinks for SEO

One of the most influential factors in search engine optimization is the content of your site's URLs. If a URL contains real words relevant to your content and target keyphrases, you are more likely to find that page in the search results.

The process of getting search engine friendly (and human friendly!) URLs is known as **URL re-writing**. By default, WordPress URLs aren't rewritten so as to avoid compatibility problems with non-Apache servers. A raw, unprocessed URL might look like this:

```
http://www.mysite.com/index.php?page=start&access=1&category=25
```

The same URL, after rewriting to improve readability and search engine effectiveness, might look like this:

```
http://www.mysite.com/about-us/company-information
```

As you can see, the second URL is a lot more readable and contains appropriate keywords that might help someone find the page. With this rewritten URL, a reader, as well as a search engine crawler, has some idea of the page's content.

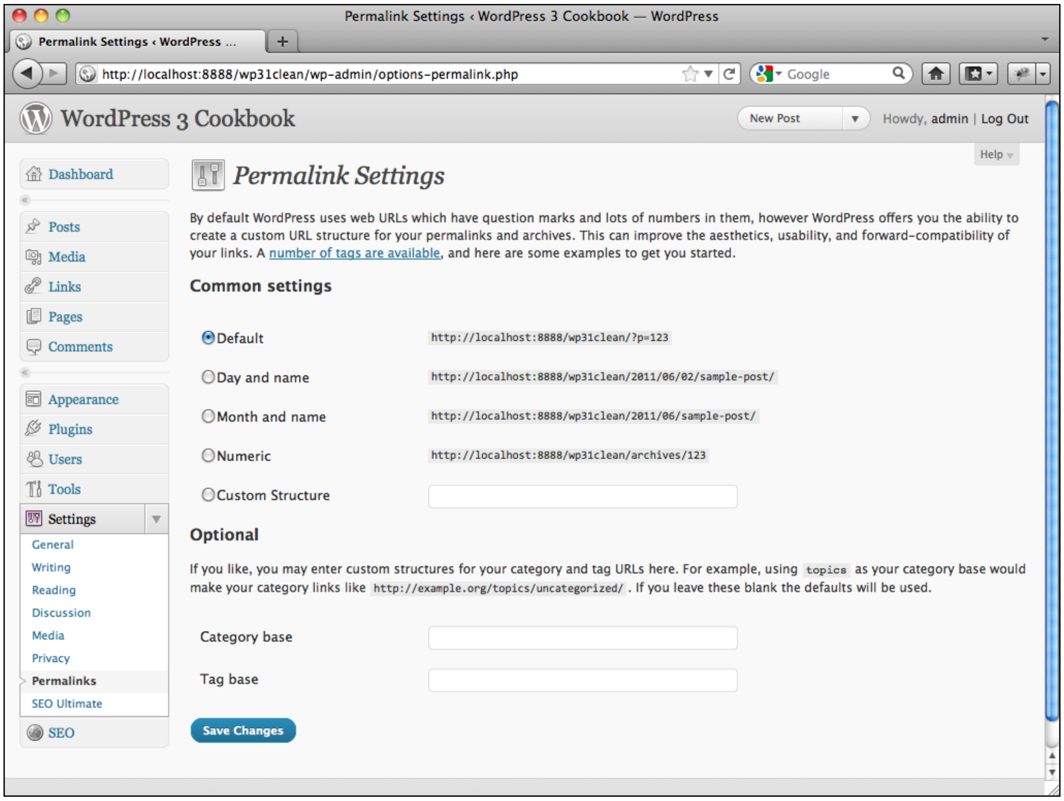
In this recipe, we show you how to implement search engine friendly URLs or, as WordPress calls them, permalinks.

Getting ready

Nothing extra is needed to cook up this recipe; everything is done from within the WordPress dashboard.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Go to **Settings**.
3. Click on the option **Permalinks**.
4. Select the permalink structure you'd like to use, or enter a custom structure.
5. Click on **Save Changes**.
6. As you can see in the following image, WordPress offers the default structure and three others predefined options:


Permalink Settings ‹ WordPress 3 Cookbook — WordPress
Permalink Settings ‹ WordPress ...
http://localhost:8888/wp31clean/wp-admin/options-permalink.php
Google
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
SEO Ultimate
SEO
Save Changes

Permalink Settings

By default WordPress uses web URLs which have question marks and lots of numbers in them, however WordPress offers you the ability to create a custom URL structure for your permalinks and archives. This can improve the aesthetics, usability, and forward-compatibility of your links. A number of tags are available, and here are some examples to get you started.

Common settings

● Default `http://localhost:8888/wp31clean/?p=123`

○ Day and name `http://localhost:8888/wp31clean/2011/06/02/sample-post/`

○ Month and name `http://localhost:8888/wp31clean/2011/06/sample-post/`

○ Numeric `http://localhost:8888/wp31clean/archives/123`

○ Custom Structure

Optional

If you like, you may enter custom structures for your category and tag URLs here. For example, using `topics` as your category base would make your category links like `http://example.org/topics/uncategorized/` . If you leave these blank the defaults will be used.

Category base

Tag base

Which permalink structure is the best? There are many online discussions about which permalink structure is the best. There is no dispositive answer to this question. While putting your keyphrases into your URLs is desirable, you must exercise a bit of restraint. Too much crammed into your URLs looks unnatural, creates clumsy URLs that are hard for people to work with, and runs the risk of being labeled deceptive by the search engines.

How it works...

The permalinks functionality relies upon the Apache mod_rewrite. The module works by transforming a URL with GET parameters to a human-readable URL. The exact structure used is dictated by the options you select on the Permalinks Settings screen.

There's more...

Now that you know how to define permalinks in your WordPress blog, let's have a look at creating custom structures for your URLs. You can also define a custom URL structure and enter it in the **Custom Structure** text field. To define a custom structure, you must enter structure tags in the Custom Structure text field. Separate each structure tag with a separator character.

Two things to keep in mind: First, never put the site URL in the custom structure fields. Second, always end with either `%post_id%` or `%postname%` to make sure the URL points to a single unique page.

Structure tags reference

WordPress gives you 10 structure tags that can be used to personalize your permalink structure. They are:

- `%author%`: The post author
- `%category%`: The post category
- `%day%`: The day the post was published, for example 03 for the third day of the month
- `%hour%`: The hour the post was published
- `%minute%`: The minute the post was published
- `%monthnum%`: The month the post was published, for example 01 for January
- `%post_id%`: The post ID, for example, 735
- `%postname%`: The name of the post
- `%second%`: The second the post was published
- `%tag%`: The tag slug field
- `%year%`: The year the post was published, for example, 2011

Structure tags can be mixed up together to create a personalized structure. Don't forget to use a separator between structure tags.

The following are the recommended separators:

- the slash (/)
- the underscore (_)
- the hyphen (-)

Further optimizing your permalinks

When you're writing a post or page, WordPress allows you to edit the permalink for the current article. You can use this tool to improve keyword density as well as for getting rid of unnecessary words in your permalinks.

For example, if you have written a post entitled "Introducing my first WordPress theme", the default permalink will be something like this:

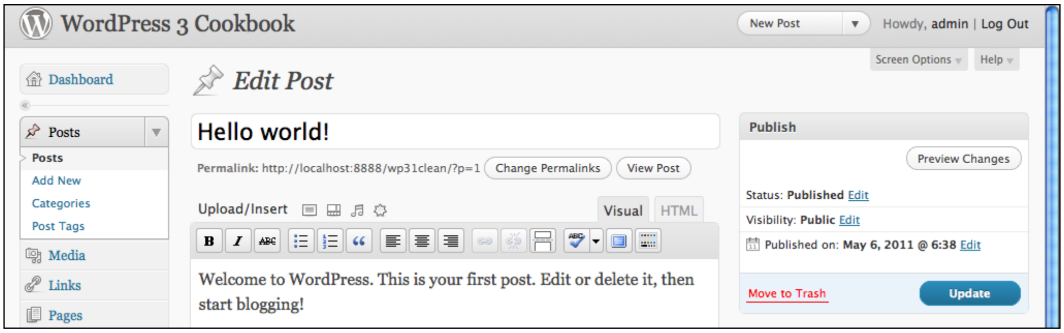
```
/introducing-my-first-wordpress-theme
```

This permalink is bad because it contains unnecessary words, and not enough keywords. Think about the keywords people will search for on Google. A much better permalink would be:

```
/free-wordpress-theme-yourthemename
```

This way, you ensure that people will be able to find your post on search engines while searching for your theme. Adding the "free" keyword is also important because many people will look for a **free** WordPress theme.

To do this, open the post for editing, then click on the **Edit** link under the post title as shown in the following screenshot. After you click on the **Edit** link, the permalink will become editable.



WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Edit Post
Posts
Posts
Add New
Categories
Post Tags
Media
Links
Pages
Publish
Preview Changes
Change Permalinks
View Post
Upload/Insert
Visual
HTML
Edit
Move to Trash
Update

Hello world!

Permalink: http://localhost:8888/wp31clean/?p=1

Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!

Status: **Published**

Visibility: **Public**

Published on: **May 6, 2011 @ 6:38**

Another use of the URL editing feature is to rid of words which aren't needed in your URL. For example, if you wrote a post entitled "Creating user-defined RSS feeds in your WordPress site", then the permalink created by WordPress would be something like this:

```
/creating-user-defined-rss-feeds-in-your-wordpress-site
```

In this permalink a few words such as "in" or "your" aren't necessary at all. Some others can be removed as well to keep your URL short and concise.

To continue with the previous example, you could edit the URL to create a tighter structure, such as:

```
creating-user-defined-rss-feeds-wordpress
```

Once done, click on the **Update** button and your changes will be recorded.

Migrating your permalinks safely

WordPress gives you the ability to modify your permalinks structure at any time. Generally, this is not a good idea as it can cause problems with inbound links from other sites and with your users' bookmarks. It is not, however, a complete disaster; if you decide that you must modify your permalinks, WordPress will help you avoid a site full of 404 errors.

Getting ready

You don't need anything special to whip up this recipe; WordPress does all the work!

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Access the permalinks settings by clicking on the option **Permalinks**.
4. Make whatever changes you want to the permalinks settings.
5. Click on **Save Changes.**

That's it, you're done. Your permalinks have been modified, but should not suffer any loss of traffic due to broken links on other websites and search engines, results pages.

How it works...

The WordPress core does the work in this recipe. The system is designed to help avoid problems when users change their permalinks structures. When you change the structure, WordPress will automatically direct traffic that comes from the old link to the proper page.

> If you are used to working with older versions of WordPress, then you will be pleased to learn that changing permalinks is no longer a recipe for problems. While, in the past, changing permalinks required a third party plugin, or a lot of work with .htaccess, it can now all be done with just the WordPress core.

Adding redirects for changed URLs

Sometimes you need to edit the URL for a specific page or post. In that situation, you run a risk of breaking links to the original page. This recipe will show you how to avoid this problem, using the Permalink Finder plugin.

Getting ready

To execute this recipe, you will need to install the Permalinks Finder plugin. You will need to install this plugin before you can get started. Search for **Permalinks Finder** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://www.blogseye.com/i-make-plugins/permalink-finder/`

How to do it...

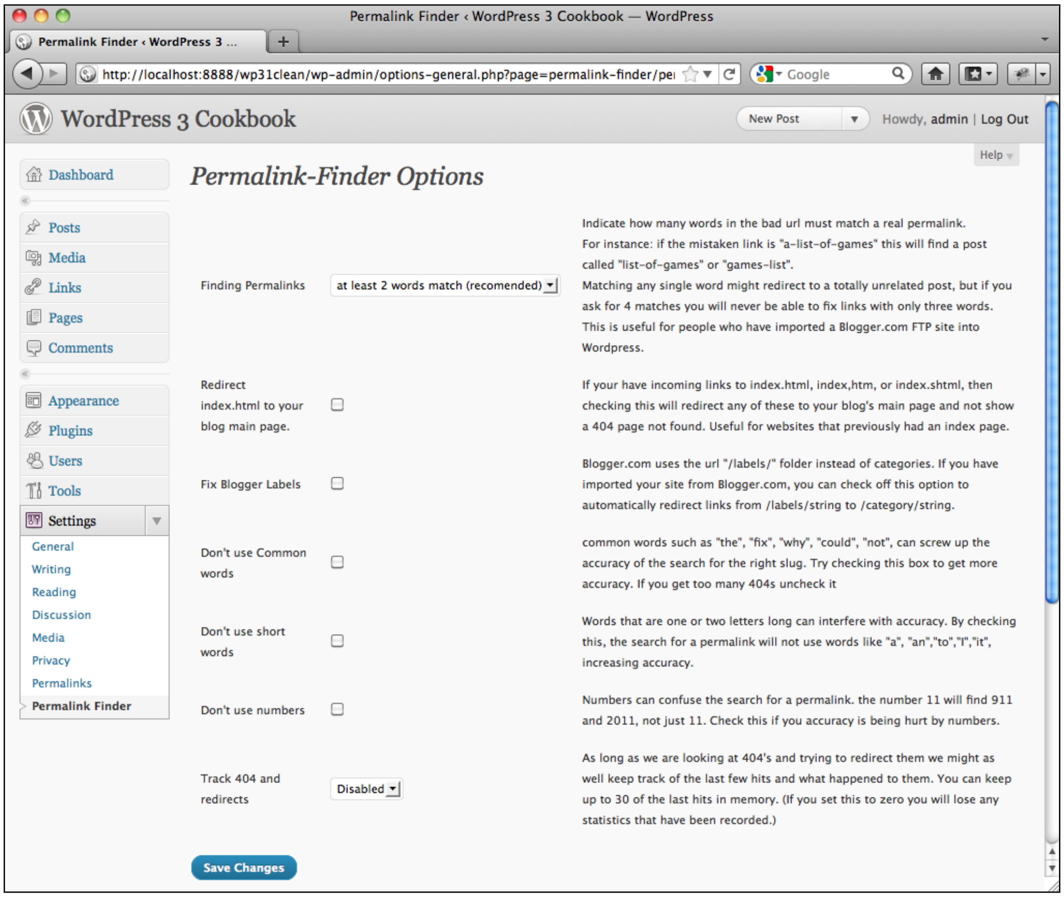
1. Click on the **Settings** menu.
2. On the **Settings** menu, click on the option **Permalink Finder**.
3. Select **Don't use Common words**.
4. Select **Don't use short words**.
5. Select **Don't use numbers**.
6. Click on **Save Changes**.

How it works...

The Permalinks Finder plugin works by detecting a request for an invalid URL, then trying to match that to an existing page. Once it finds the proper page, it will send the user to the right page. Additionally, the plugin sends a 301 redirect to help inform search engines and spiders that the link has changed.

There's more...

The Permalink Finder plugin is also very useful to you if you have migrated an existing website to WordPress. By installing and activating the plugin, you can prevent loss of traffic to both internal pages and the home page of your new WordPress site. The plugin also includes a specific option intended to help those that are migrating from Blogger to WordPress, as you can see in the following screenshot:



Permalink Finder ‹ WordPress 3 Cookbook — WordPress
Permalink Finder ‹ WordPress 3 ...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=permalink-finder/pe
Google
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
Permalink Finder
Save Changes

Permalink-Finder Options

Finding Permalinks	at least 2 words match (recomended)	Indicate how many words in the bad url must match a real permalink. For instance: if the mistaken link is "a-list-of-games" this will find a post called "list-of-games" or "games-list". Matching any single word might redirect to a totally unrelated post, but if you ask for 4 matches you will never be able to fix links with only three words. This is useful for people who have imported a Blogger.com FTP site into Wordpress.
Redirect index.html to your blog main page.	☐	If your have incoming links to index.html, index,htm, or index.shtml, then checking this will redirect any of these to your blog's main page and not show a 404 page not found. Useful for websites that previously had an index page.
Fix Blogger Labels	☐	Blogger.com uses the url "/labels/" folder instead of categories. If you have imported your site from Blogger.com, you can check off this option to automatically redirect links from /labels/string to /category/string.
Don't use Common words	☐	common words such as "the", "fix", "why", "could", "not", can screw up the accuracy of the search for the right slug. Try checking this box to get more accuracy. If you get too many 404s uncheck it
Don't use short words	☐	Words that are one or two letters long can interfere with accuracy. By checking this, the search for a permalink will not use words like "a", "an","to","I","it", increasing accuracy.
Don't use numbers	☐	Numbers can confuse the search for a permalink. the number 11 will find 911 and 2011, not just 11. Check this if you accuracy is being hurt by numbers.
Track 404 and redirects	Disabled	As long as we are looking at 404's and trying to redirect them we might as well keep track of the last few hits and what happened to them. You can keep up to 30 of the last hits in memory. (If you set this to zero you will lose any statistics that have been recorded.)

Creating meta descriptions for your posts and pages

By default, WordPress doesn't add the meta description tag to the posts you write. While meta descriptions aren't as important as they used to be in SEO, they are always a plus for both your visitors and search engine bots. You do want to include these.

In this recipe we look at several solutions for providing your site with a meta description field. In the first part of this recipe, we look at two options for creating static (unchanging) descriptions. In the latter part of the recipe we look at a more advanced solution that lets you edit the description for individual posts.

Getting ready

To complete this recipe, you will need access to your WordPress installation on your server and you will also need a text editor. To create the more advanced solution, discussed in the There's more section, you will need to install the SEO Ultimate plugin, discussed later in this chapter.

How to do it...

Let's start by looking at two static solutions, that is, solutions that set a static description that will be applied across all pages. In the first option, you are simply adding a line of code to your theme that specifies the meta description tag's content throughout the site. In the second option, we simply use the text input for the Tagline in the General Settings in the administration system of your WordPress installation. Using the latter method, if you wish to change the meta description for the site, you can do so by simply editing the Tagline value inside the dashboard.

1. Access the files for your WordPress installation on your server.
2. Open `header.php` for editing – use the file inside your active theme.
3. If you wish to add a standardized meta description for use on all pages, you can do so by adding the following code somewhere inside the `<head>` of the file:

   ```
   <meta name="description" content="the description you wish to
   use will go here, inside the quotations" />
   ```

4. If you wish to use the site's description, which is editable from within the admin system, use this code instead:

   ```
   <meta name="description" content="<?php
   bloginfo('description'); ?>" />
   ```

5. Save the `header.php` file.

Meta descriptions are now displayed in the head of your WordPress posts and pages.

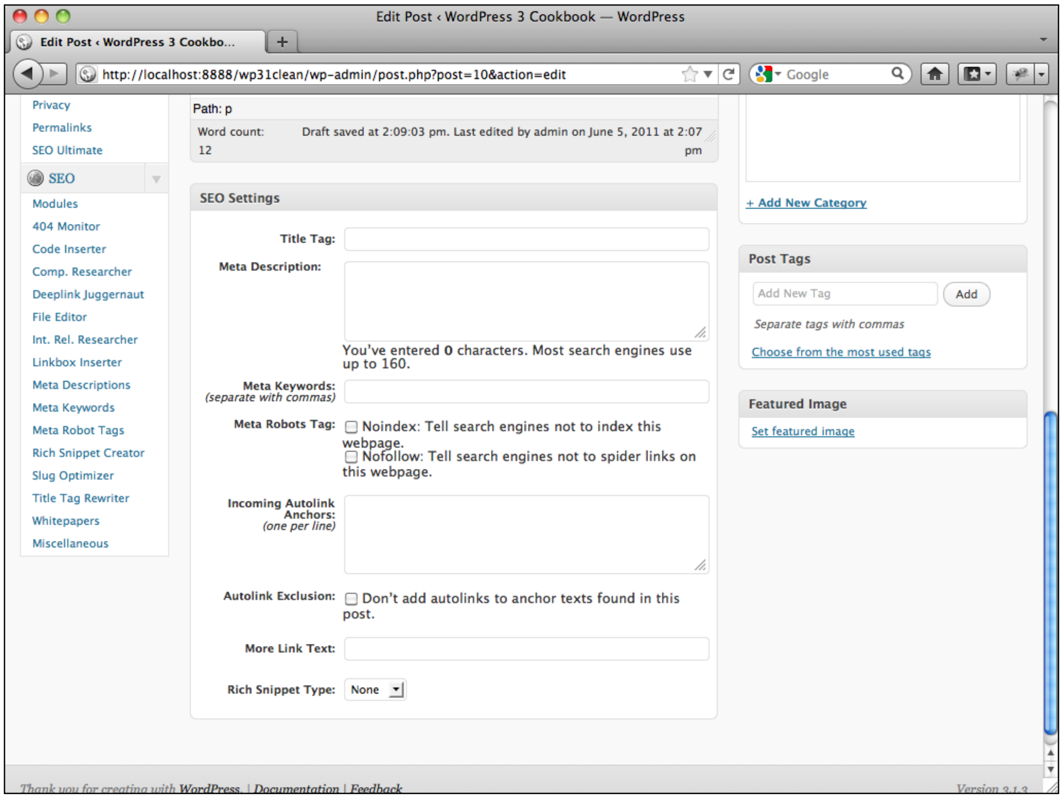
How it works...

In the first example, the value for the meta description tag is simply hard coded. In the second example, the value input by the administrator for the site's tagline is automatically printed on the page as the value for the meta description field.

There's more...

While the preceding code is good, it is very limited. It is not ideal to have the same meta description value for all your pages; it would be better if you could control the description on individual pages so that the description tag can be tailored to match the contents of the page.

To achieve a more flexible solution, install the SEO Ultimate plugin, discussed later in this chapter. Once the plugin is installed and activated, each of the posts on your site will now have a new **SEO Settings** section, as shown in thefollowing screenshot:


Edit Post ‹ WordPress 3 Cookbook — WordPress
Edit Post ‹ WordPress 3 Cookbo...
http://localhost:8888/wp31clean/wp-admin/post.php?post=10&action=edit
Google
Privacy
Permalinks
SEO Ultimate
SEO
Modules
404 Monitor
Code Inserter
Comp. Researcher
Deeplink Juggernaut
File Editor
Int. Rel. Researcher
Linkbox Inserter
Meta Descriptions
Meta Keywords
Meta Robot Tags
Rich Snippet Creator
Slug Optimizer
Title Tag Rewriter
Whitepapers
Miscellaneous
+ Add New Category
Post Tags
Add New Tag
Add
Separate tags with commas
Choose from the most used tags
Featured Image
Set featured image
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

Path: p

Word count: 12 Draft saved at 2:09:03 pm. Last edited by admin on June 5, 2011 at 2:07 pm

SEO Settings

Title Tag:

Meta Description:

You've entered 0 characters. Most search engines use up to 160.

Meta Keywords: (separate with commas)

Meta Robots Tag: ☐ Noindex: Tell search engines not to index this webpage.
☐ Nofollow: Tell search engines not to spider links on this webpage.

Incoming Autolink Anchors: (one per line)

Autolink Exclusion: ☐ Don't add autolinks to anchor texts found in this post.

More Link Text:

Rich Snippet Type: None

The fields include not only a meta description option, but also a title tag and other options.

See also

- *Improving SEO with the SEO Ultimate plugin*

Avoiding duplicate content with a robots.txt file

To avoid being penalized for duplicate content on your site, you can use a simple file tailored to that task. This useful file is a basic text file, named `robots.txt` and it is located at the root of your WordPress blog. The file is used to tell the search engines crawlers that they don't have to follow or index some pages or directories.

Getting ready

To implement this recipe you will need a text editor and access to the server where your WordPress site is installed.

How to do it...

1. Create a new blank file on your computer. Name it `robots.txt`.
2. Copy and paste the following code in the `robots.txt` file:

```
User-agent: *
Disallow: /wp-
Disallow: /search
Disallow: /feed
Disallow: /comments/feed
Disallow: /feed/$
Disallow: /*/feed/$
Disallow: /*/feed/rss/$
Disallow: /*/trackback/$
Disallow: /*/*/feed/$
Disallow: /*/*/feed/rss/$
Disallow: /*/*/trackback/$
Disallow: /*/*/*/feed/$
Disallow: /*/*/*/feed/rss/$
Disallow: /*/*/*/trackback/$
```

3. Save the file.
4. Upload it to your WordPress site root.

> Obviously this recipe assumes you don't already have a `robots.txt` file on your site. If you do, edit that file, rather than overwriting it completely with a new one!

How it works...

This `robots.txt` file prevents duplicate content by telling all search engines crawlers not to index search results, feeds, trackbacks URLs, as well as `wp-*` directories.

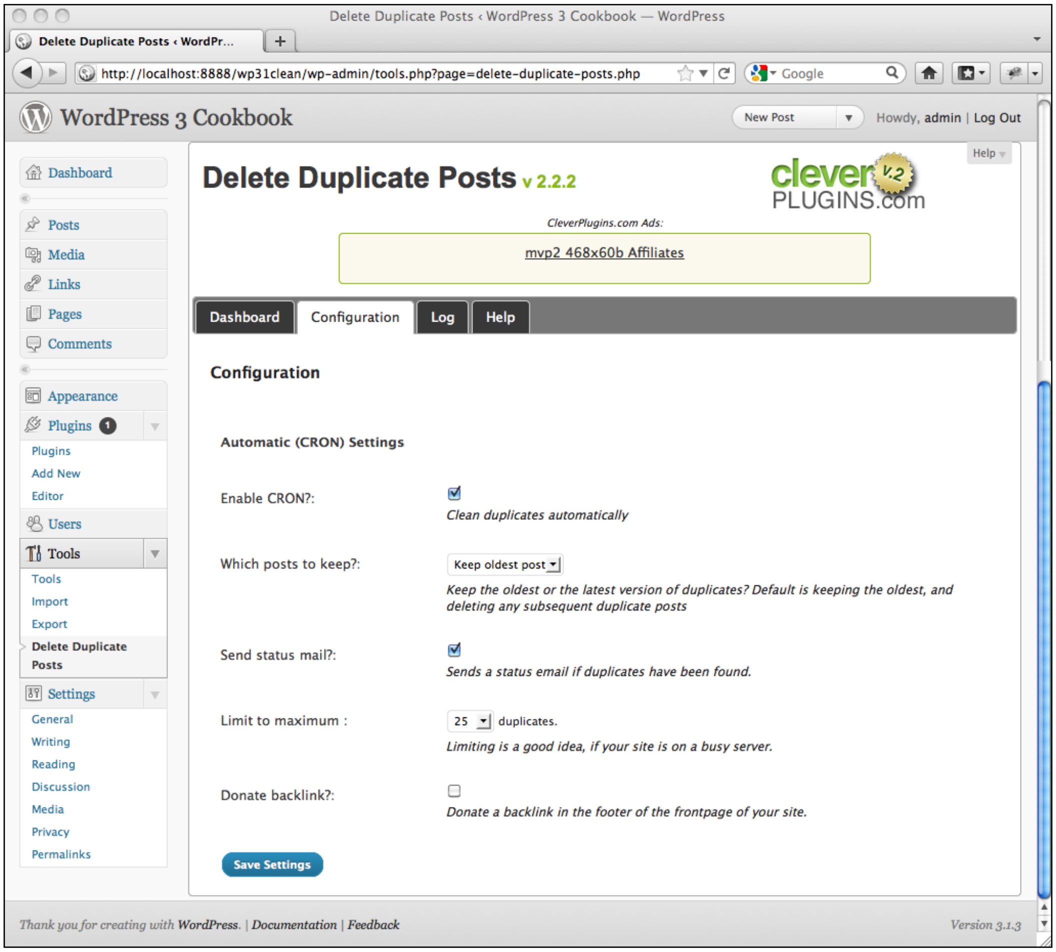
There's more...

The `robots.txt` file is the most common technique used to avoid duplicate content on web sites. Though, as you can guess, this isn't the only way to protect your WordPress blog from search engine optimization problems caused by duplicate content.

A plugin to remove duplicate content

While the `robots.txt` recipe above is effective at preventing the double indexing of the content on your site, there are other ways that your site may find itself with duplicate content. If you have saved multiple copies of the same post or page during editing, or if you are using a scraper or RSS aggregator, it is possible that multiple copies of the same post may wind up in your site; if those circumstances arise, the `robots.txt` fix outlined above will not solve the problem.

The solution is either to diligently monitor your site, or install an automated tool to watch it for you. The **Delete Duplicate Posts** plugin is one solution. The plugin will monitor your site each cron run, search for duplicate content, then eliminate the duplicates. The settings for the plugin, as shown in the following screenshot, allow you to set automatic or manual cleaning, determine the deletion criteria, and even set up e-mail notifications when duplicate posts are found:


Delete Duplicate Posts ‹ WordPress 3 Cookbook — WordPress
http://localhost:8888/wp31clean/wp-admin/tools.php?page=delete-duplicate-posts.php
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Add New
Editor
Users
Tools
Import
Export
Delete Duplicate Posts
Settings
General
Writing
Reading
Discussion
Privacy
Permalinks
clever PLUGINS.com
CleverPlugins.com Ads:
mvp2 468x60b Affiliates
Dashboard
Configuration
Log
Help
Save Settings
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

Delete Duplicate Posts v 2.2.2

Configuration

Automatic (CRON) Settings

Enable CRON?: ☑ Clean duplicates automatically

Which posts to keep?: Keep oldest post
Keep the oldest or the latest version of duplicates? Default is keeping the oldest, and deleting any subsequent duplicate posts

Send status mail?: ☑ Sends a status email if duplicates have been found.

Limit to maximum : 25 duplicates.
Limiting is a good idea, if your site is on a busy server.

Donate backlink?: ☐ Donate a backlink in the footer of the frontpage of your site.

Pinging third-party services

If you're interested in SEO, then you are interested in getting more visitors. A good way to bring new readers is to ping third-party services.

In this recipe, we're looking at third-party services, such as Technorati or Google blog search. These services publish links to popular posts and allow visitors to search among blogs.

Pinging is simply the act of notifying these services that new content is available on your site. After you have successfully pinged for example, Technorati, a crawler from that site will visit your site and index your newest content, making it available to their visitors.

Getting ready

To ping a website, you have to get the trackback URL. For example, Technorati trackback URL is `http://rpc.technorati.com/rpc/ping`.

> Finding trackback or ping URLs isn't always easy as most websites just display a link on their footer, or even worse, on a specific page. Often the quickest way to find the ping URL is to run a Google search. To perform this search quickly and accurately, use the following specific syntax:
>
> `site:technorati.com ping url`

How to do it...

Once you have the ping URLs of the websites you'd like to notify about your new content, you have to configure WordPress so that it will automatically ping the selected website when you write new posts or pages.

To do so, follow this short procedure:

1. Log in to your WordPress **Dashboard.**
2. Go to **Settings.**
3. Click on **Writing**.
4. Scroll down to the bottom of the page. You'll see the **Update Services** paragraph, which includes a text area in which you can specify one or more URLs you'd like to ping.
5. Paste your ping URLs in the text area. Make sure to write each URL on a new line.
6. Save your URLs by clicking on the **Save Changes** button.

That's all it takes. The next time you publish a new post, your WordPress system will automatically ping the sites and notify them of the new content.

How it works...

To notify other websites about the changes that you made in yours, WordPress uses the **RPC** (**Remote Procedure Call**) technology that consists of an inter-process communication technology to send a signal to another web site. When a web site (one that accepts pings) receives a ping, it automatically executes a bunch of functions. For example, it can tell a crawler to visit and index the site which has pinged it, or it can post a comment to the blog. This is exactly how trackbacks or pingbacks work.

[The WordPress Codex includes a list of recommended Ping URLs and related services. Visit `http://codex.wordpress.org/Update_Services`]

Enhancing site indexing with XML sitemaps

While it is one thing to create a website and build pages of content, it is another thing entirely to make sure that all of those pages are indexed by the search engines. The simple act of creating a web page does not assure that it will be indexed. While the search engines today are much better at detecting and indexing new content, it is best to provide them with some guidance and to help assure that all the pages are indexed in a timely fashion. The best way to guarantee that the search engine crawlers can see and index all of your site pages is to use an XML sitemap.

According to `sitemaps.org`, "sitemaps are an easy way for webmasters to inform search engines about pages on their sites that are available for crawling. In its simplest form, a sitemap is an XML file that lists URLs for a site along with additional metadata about each URL (when it was last updated, how often it usually changes, and how important it is, relative to other URLs in the site) so that search engines can more intelligently crawl the site."

In other words, although using a sitemap doesn't guarantee that absolutely all your pages will be indexed by search engines, it does gives crawlers some information to do a better job. All three of the major search engines follow the XML sitemaps protocol and, as you will see in a later recipe, Google and Bing provide services to help you monitor the effectiveness of the sitemaps.

[Google and Bing Webmaster services are discussed in the next recipe in this chapter.]

Sitemaps can be created manually, but that approach requires you to update the sitemap file after each new post or page is published. The better course is to use a third-party plugin that's purpose-built for managing your XML sitemaps. In this recipe, we look at implementing the **XML Sitemap Generator** for WordPress. At the end of this recipe, you will have an XML sitemap that automatically updates to reflect your content changes and notifies the search engines that there are new pages to index.

Getting ready

XML Sitemap Generator is a third-party plugin. Accordingly, you must first install and enable this plugin on your WordPress site. Search for **XML Sitemap Generator** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about this plugin by visiting the developer's site at `http://www.arnebrachhold.de/projects/wordpress-plugins/google-xml-sitemaps-generator/`

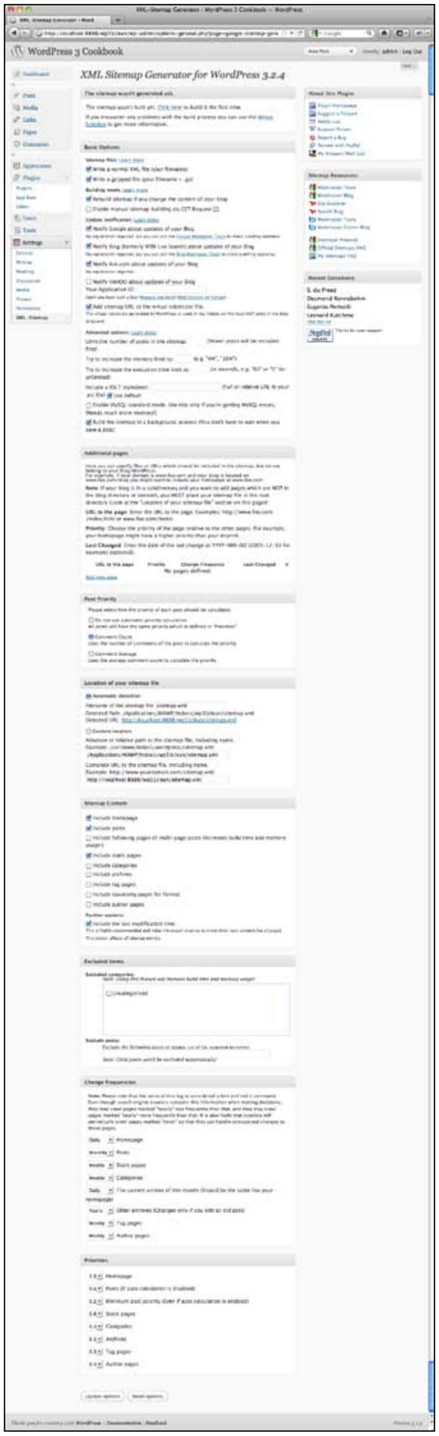
How to do it...

1. Log in to your WordPress **Dashboard**.
2. Go to the **Settings** tab.
3. Click on the option **XML-Sitemap**.
4. On the **XML Sitemap Generator Settings** page you can select from multiple options, as shown in the next screenshot. The default settings will be fine for most sites.

> Note: If you are using a dedicated `robots.txt` file, such as the one recommended earlier in this chapter, de-select the option **Add sitemap URL to the virtual robots.txt file**.

5. After you have made your selections, click on the **Update options** button.

If you want to check the result, simply open a browser and visit `http://www.yoursite.com/sitemap.xml` and you should be able to see your new sitemap. If the output is not what you expected, repeat the steps above to adjust the settings.


WordPress 3 Cookbook
XML Sitemap Generator for WordPress 3.2.4

How it works...

A sitemap is, as an RSS feed, a good old XML file, encoded in UTF-8. Here's an example of a sitemap:

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
<url>
    <loc>http://www.example.com/</loc>
    <lastmod>2005-01-01</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.8</priority>
</url>
<url>
    <loc>http://www.example.com/catalog?item=73&amp;desc=
        vacation_new_zealand</loc>
    <lastmod>2004-12-23</lastmod>
    <changefreq>weekly</changefreq>
</url>
<url>
<loc>http://www.example.com/catalog?item=74&amp;desc=
    vacation_newfoundland</loc>
    <lastmod>2004-12-23T18:00:15+00:00</lastmod>
    <priority>0.3</priority>
</url>
<url>
    <loc>http://www.example.com/catalog?item=83&amp;desc=
        vacation_usa</loc>
    <lastmod>2004-11-23</lastmod>
</url>
</urlset>
```

Let's have a look at the different attributes of this standard sitemap:

- `urlset`: Reference the sitemap protoc (required)
- `urn`: This is the parent tag for each URL of your site (required)
- `loc`: The URL of the page; must start with `http://` (required)
- `lastmod`: The date of the last modification on the page; format must be YYYY-MM-DD (optional)
- `changefreq`: How frequently the content of the page is updated; values can be `always`, `hourly`, `daily`, `weekly`, `monthly`, `yearly`, or `never` (optional)
- `priority`: The importance of this URL relative to other URLs from your site; values range from 0.0 to 1.0 and the default priority is 0.5 (optional)

If you decide to create your own, you will need to follow a few important rules:

- Sitemaps must have an `.xml` extension.
- Sitemaps must be encoded using UTF-8.
- All data in a sitemap must be entity-escaped. For example, you can't include the less than character (`<`). You have to use the related escaped entity, which is `&lt;`.
- A sitemap must always begin with the `<urlset>` tag, and close with the corresponding closing `</urlset>` tag.
- `<url>` and `<loc>` tags are mandatory.

See also

- *Using Google's and Bing's Webmaster Tools*

Using Google's and Bing's Webmaster Tools

Now that you have created an XML sitemap, you should also consider taking advantage of two services maintained by Google and Bing. Both services have similar names and provide similar functionality: Google Webmaster Central and Bing Webmaster Tools.

These popular services provide a place for website owners to gain insight into how their websites are performing on the search engines. Both systems provide a set of tools and reports that allow you to see how your site is ranked and what you can do to enhance it.

Both systems require you to first verify your site. While you can verify manually, by adding a file to your server, or a tag to your page code, in this recipe we explore an easy-to-use plugin that can make verification easier. The plugin is called the All in One Webmaster and it handles Google, Bing, and a number of other services as well.

Getting ready

All in One Webmaster is a third-party plugin. Accordingly, you must first install and enable this plugin on your WordPress site. Search for **All in One Webmaster** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about this plugin by visiting the developer's site at `http://arpitshah.com/post/10117563852/all-in-one-webmaster`

Next, if you have not done so already, register with Google and with Bing to gain access to their webmaster tools. You will need to have access to these services to complete this recipe.

How to do it...

Let's start with Google Webmaster Central:

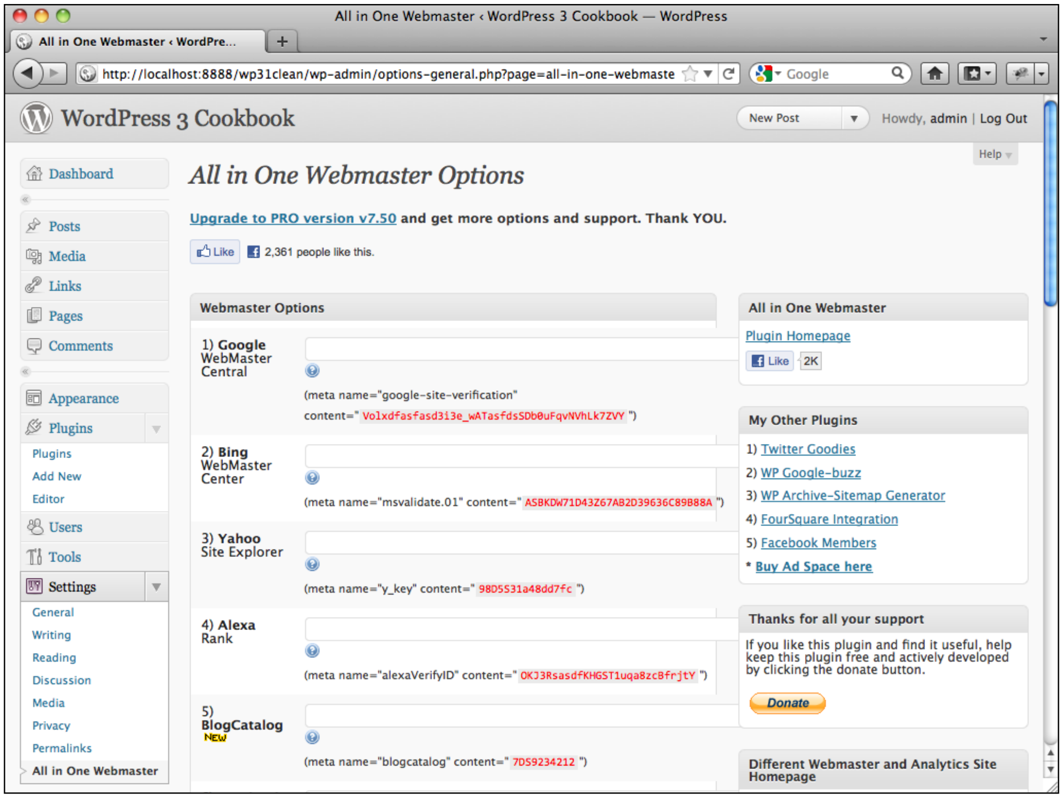
1. Log in to Google Webmaster Central by visiting `https://www.google.com/webmasters`
2. Click on the button labeled **Add a site**.
3. In the popup that appears, add your site's URL and then click on **Continue**.
4. You must next verify your site. Google will give you a choice of verification methods, but for this recipe we're using the **All in One Webmaster** plugin, so click on the **Alternate methods** tab on the **Verify ownership** screen.
5. Click on the option **Add a meta tag** to your site's home page.
6. Find the verification number in the code that is shown – it will look something like this: **f51uBC2isisNfm22KeaS7PGWHmqYn66BezUSTTCCfiM**.
7. Copy the number and leave the Google Webmaster Central page open in your browser while you perform the next steps.
8. Log in to your WordPress admin dashboard.
9. Click on the **Settings** menu.
10. Click on the option **All in One Webmaster**.
11. On the screen that loads, paste the Google verification code into the blank field labeled **Google WebMaster Central**, as you can see in the screenshot below.
12. Click on **Update options**.
13. Return to Google Webmaster Central, and click on the **Verify** button.

If you were successful, you will see a confirmation message.

Next, let's set up Bing's Webmaster Tools:

1. Log in to Google Webmaster Central by visiting `http://www.bing.com/toolbox/webmasters/`
2. Click on the button labeled **Add site**.
3. In the popup that appears, add your site's URL and then click on **Submit**.
4. You must next verify your site. Bing will give you a choice of verification methods, but for this recipe we're using the **All in One Webmaster** plugin, so click on **Option 2: Copy and paste a tag in your default webpage**.
5. Find the verification number in the code that is shown – it will look something like this: **C27D268A4736287D9264RG7VV50Z401B**.
6. Copy the number and leave the **Bing Webmaster Tools** page open in your browser while you perform the next steps.

7. Log in to your WordPress admin dashboard.
8. Click on the **Settings** menu.
9. Click on the option **All in One Webmaster**.
10. On the screen that loads, paste the Google verification code into the blank field labeled **Bing WebMaster Central**, as you can see in the following screenshot:


All in One Webmaster ‹ WordPress 3 Cookbook — WordPress
All in One Webmaster ‹ WordPre...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=all-in-one-webmaste
Google
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
All in One Webmaster
Like
Plugin Homepage
Like
2K
Donate

All in One Webmaster Options

Upgrade to PRO version v7.50 and get more options and support. Thank YOU.

2,361 people like this.

Webmaster Options

1) **Google** WebMaster Central

(meta name="google-site-verification" content=" Volxdfasfasd3i3e_wATasfdsSDb0uFqvNVhLk7ZVY ")

2) **Bing** WebMaster Center

(meta name="msvalidate.01" content=" ASBKDW71D43Z67AB2D39636C89B88A ")

3) **Yahoo** Site Explorer

(meta name="y_key" content=" 98D5S31a48dd7fc ")

4) **Alexa** Rank

(meta name="alexaVerifyID" content=" OKJ3RsasdfKHGST1uqa8zcBfrjtY ")

5) **BlogCatalog** NEW

(meta name="blogcatalog" content=" 7DS9234212 ")

All in One Webmaster

My Other Plugins

1) Twitter Goodies
2) WP Google-buzz
3) WP Archive-Sitemap Generator
4) FourSquare Integration
5) Facebook Members
* Buy Ad Space here

Thanks for all your support

If you like this plugin and find it useful, help keep this plugin free and actively developed by clicking the donate button.

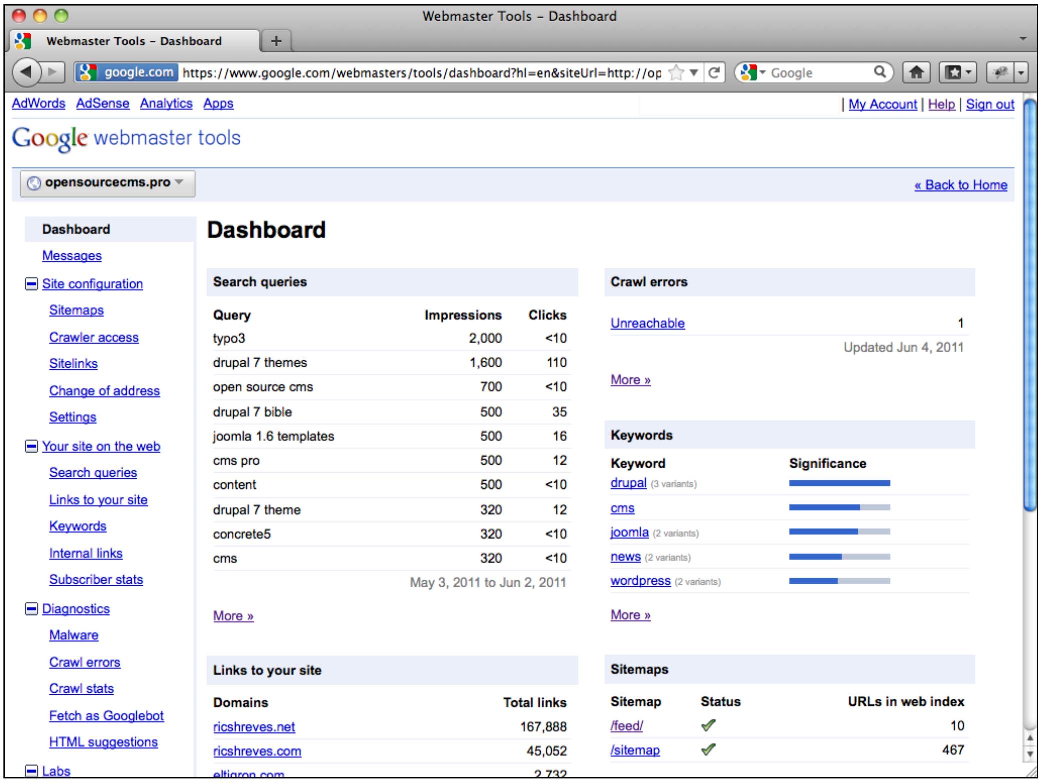
Different Webmaster and Analytics Site Homepage

11. Click on **Update options**.
12. Return to **Bing Webmaster Tools**, and click on the **Verify** button.
13. If you were successful, you will see a confirmation message.

How it works...

These two services work in a very similar fashion and provide very similar tools. Let's look at Google Webmaster Central for an example of what can be done.

Once you've verified your site and Google has a chance to index it, you can log in to your Google account and access all tools and reports.


Webmaster Tools – Dashboard
Webmaster Tools – Dashboard
google.com
https://www.google.com/webmasters/tools/dashboard?hl=en&siteUrl=http://op
Google
AdWords AdSense Analytics Apps
| My Account | Help | Sign out
Google webmaster tools
opensourcecms.pro
« Back to Home
Dashboard
Messages
Site configuration
Sitemaps
Crawler access
Sitelinks
Change of address
Settings
Your site on the web
Search queries
Links to your site
Keywords
Internal links
Subscriber stats
Diagnostics
Malware
Crawl errors
Crawl stats
Fetch as Googlebot
HTML suggestions
Labs

Dashboard

Search queries

Query	Impressions	Clicks
typo3	2,000	<10
drupal 7 themes	1,600	110
open source cms	700	<10
drupal 7 bible	500	35
joomla 1.6 templates	500	16
cms pro	500	12
content	500	<10
drupal 7 theme	320	12
concrete5	320	<10
cms	320	<10

May 3, 2011 to Jun 2, 2011

More »

Crawl errors

Unreachable	1

Updated Jun 4, 2011

More »

Keywords

Keyword	Significance
drupal (3 variants)	
cms	
joomla (2 variants)	
news (2 variants)	
wordpress (2 variants)	

More »

Links to your site

Domains	Total links
ricshreves.net	167,888
ricshreves.com	45,052
eltiqron.com	2,732

Sitemaps

Sitemap	Status	URLs in web index
/feed/	✓	10
/sitemap	✓	467

There's more...

Once your site is verified, you can access all of the powerful features and reports provided by Google Webmaster Tools.

Site configuration

This section of Webmaster Central provides you with an overview of the status of your site in the Google index. You can learn whether your feeds and sitemap are being indexed and you can also evaluate whether your `robots.txt` file is appropriate and not creating problems for the Google spider.

Your site on the web

The Your site on the web section shows how often your site is appearing in the search results and how many clicks came from Google searches. You can also review a list of keywords where your site has appeared in searches and a list of your internal links.

Diagnostics

As the name implies, the Diagnostics section helps you identify whether there are any significant problems with your site – from Google's perspective. Reports indicate whether Google suspects there is malware on our site and more importantly, gives you insights into crawl errors and statistics. This section also gives you a tool that lets you view your site as Google sees it, which is useful in helping to identify whether you key meta data and contents are visible to the Google spider.

Labs

Labs contains a varying set of new tools in development at Google Labs. Check this section regularly as the contents do change and sometimes you can find some very interesting tools that are still largely new to the Google world.

See also

- *Enhancing site indexing with XML sitemaps*

Improving SEO with the SEO Ultimate plugin

SEO is a popular topic and there are many WordPress plugins dedicated to enhancing your site's SEO. One of the best of the group is SEO Ultimate, a plugin that includes a large number of features to help optimize your site for the search engines. Among the functionality included with this plugin:

- Generate META tags automatically
- Edit META tags throughout the site
- Monitor for 404s
- Maintain canonical URLs
- Handle robot tags
- Insert code into the top or bottom of your pages
- Automatically create internal links

The plugin also includes some things you might not expect, including a competitor analysis tool and the ability to automatically insert a linkbox at the end of each content item.

Getting ready

SEO Ultimate is a third-party plugin. Accordingly, you must first install and enable this plugin on your WordPress site. Search for **SEO Ultimate** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> To learn more about this plugin, visit the developer's site at `http://www.seodesignsolutions.com/wordpress-seo/`

How to do it...

After you have successfully installed the **SEO Ultimate** plugin, you will want to configure it a bit. To do so, follow these simple steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **SEO** menu.
3. On the page that appears, select the modules you wish to use; disable those you do not intend to use.
4. Among the key modules you will want to configure are the following:
    - **Meta Descriptions**
    - **Meta Keywords**
    - **Meta Robot Tags**
    - **Slug Optimizer**
    - **Title Tag Rewriter**
5. As you configure each of these options, click on the **Save Changes** button after you finish each.

There's more...

Earlier in this chapter we discussed creating custom page titles and meta descriptions. Those recipes give both a manual option for dealing with these areas and the option to use the SEO Ultimate plugin. For those who want the most control, the SEO Ultimate option is the better choice. If you wish to use the manual approach, then you should disable the relevant modules in the plugin.

Five more tips for a better SEO

In this chapter, we saw many tips and tools to enhance your site's SEO and get more traffic from the search engines. SEO is a big subject and there are many other things that you can do to improve your likelihood of success. Here are five quick tips to help you get better rankings.

Get backlinks

In terms of SEO, the number of links back to your site is a key factor. Basically, the more backlinks you have, the better you'll be ranked. Getting backlinks isn't always easy. If, however, you produce good content, other sites will link to you. If you want to be more aggressive about building links, seek out similar sites and contact them directly to propose exchanging links.

Use the proper h tags structure

Proper use of `h` tags is important both for SEO and for accessibility. The tags should be used consistently, and in a fashion that clearly indicates the hierarchical structure of the content item.

An example of a bad title would be this:

```
<strong><font color="#00FF00" size="24px">My blog post
    title</font></strong>
```

A better title would look like this:

```
<h2>My blog post title</h2>
```

With the following in your `style.css` file:

```
H2 {font-family: Arial; size: 24px;
color: #00FF00; font-weight: bold;}
```

Both of these code examples visually appear the same, but the second is much better for SEO purposes. Do not hesitate to check your theme file and verify that they're using a proper `h` tag structure.

Make sure your blog is XHTML valid

Search engines love valid XHTML websites. Having a valid XHTML codeset means that your pages are coded according to the W3C recommendations.

To check if your site is valid in terms of XHTML, go to `http://validator.w3.org/` and type your site's URL. If you have any errors, the validator will tell you where they're located and why it is a problem.

Use keywords in your content items

An important point for search engine ranking is keyword density. This refers to the frequency that a term appears throughout the code for a specific page.

Let's say you wrote a post about SEO for WordPress blogs. You'll probably want your post to be featured in search engines results pages when someone searches for keywords as such as 'WordPress SEO', or 'blog SEO'.

To improve your odds for success, you should start by writing a keyword-rich title for the article. A good example would be **SEO for WordPress blogs** or **Tips and tricks for better WordPress SEO**. These are good titles because they are concise, descriptive, and contain appropriate keywords.

It is also important that your post content contains the same keywords. You don't need to repeat them on each paragraph of course, but make sure to use your keywords frequently. You should also make sure the meta tags for page reflect the same keywords. Taken together, when managed properly, these factors will give you appropriate keyword density and improve your odds of ranking well for that page.

See also

- *Creating meta descriptions for your posts and pages*

8

Enhancing Usability and Accessibility

In this chapter you will learn about:

- Creating print friendly pages
- Extending WordPress search
- Enhancing navigation with breadcrumbs
- Stopping SPAM
- Optimizing performance with cache management
- Displaying a login form
- Displaying related posts
- Creating a Feature Posts block
- Adding a sitemap for your site visitors
- Creating a better tag cloud
- Adding lightboxes for your images

Introduction

Good site design includes putting thought into how to make the site easy to use to the widest number of people. In this chapter, we look at a number of plugins and techniques that are intended to make your site friendlier for visitors.

The recipes in this chapter are concerned with improving navigation, and making it easier for people to discover popular and relevant content. Usability is, in other words, not only good for the users, but also good for the site owners, as a usable site typically translates into more page views and better user engagement.

In this chapter, we also look at some enhancement techniques that are popular in site design today, for example, implementing tag clouds and using lightboxes for image display. While none of the recipes in this chapter are absolutely essential for your site, it is definitely worth considering adding at least some of these to your site.

Creating print-friendly pages

Yes, it may seem a little obsolete in a technically advanced world and it isn't green at all, but the fact is that many Internet users still print pages for offline consultation. While some social sharing plugins also offer print options and you can always use your browser's print function, you may want to have more control over what your pages look like when printed.

In this recipe we install a plugin that not only enables a print friendly option, but also gives you complete control over the output.

Getting ready

To execute this recipe, you will need to install the **WP Print Friendly** plugin. You will need to install this plugin before you can get started. Search for **WP Print Friendly** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://www.thinkoomph.com/plugins-modules/wp-print-friendly/`

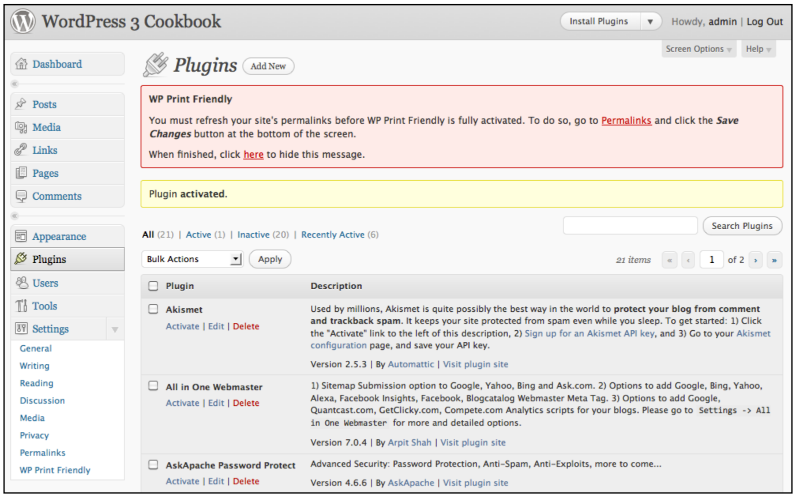
If you wish to apply any of the more advanced techniques discussed later in this recipe, you will also need a code editor and access to the files of the WordPress installation on your server.

> This plugin has been known to cause conflicts with some SEO and permalinks plugins. Consult the plugin's FAQ page on WordPress.org to view an updated list.

How to do it...

Let us start with getting the system ready and configuring the plugin:

1. Log in to the WordPress **Dashboard**.
2. You will see a notification at the top of the page, advising you to refresh your permalinks, as shown in the following screenshot:


WordPress 3 Cookbook
Install Plugins
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
WP Print Friendly
Plugins
Add New
Search Plugins
Bulk Actions
Apply

WP Print Friendly

You must refresh your site's permalinks before WP Print Friendly is fully activated. To do so, go to Permalinks and click the ***Save Changes*** button at the bottom of the screen.

When finished, click here to hide this message.

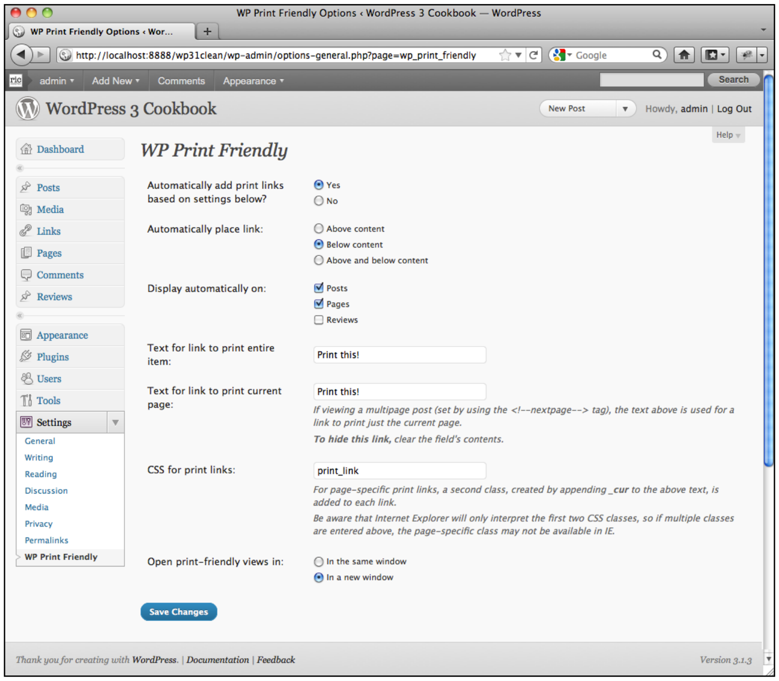
Plugin **activated**.

All (21) | Active (1) | Inactive (20) | Recently Active (6)

21 items « ‹ 1 of 2 › »

☐ Plugin	Description
☐ **Akismet** Activate | Edit | Delete	Used by millions, Akismet is quite possibly the best way in the world to **protect your blog from comment and trackback spam**. It keeps your site protected from spam even while you sleep. To get started: 1) Click the "Activate" link to the left of this description, 2) Sign up for an Akismet API key, and 3) Go to your Akismet configuration page, and save your API key. Version 2.5.3 | By Automattic | Visit plugin site
☐ **All in One Webmaster** Activate | Edit | Delete	1) Sitemap Submission option to Google, Yahoo, Bing and Ask.com. 2) Options to add Google, Bing, Yahoo, Alexa, Facebook Insights, Facebook, Blogcatalog Webmaster Meta Tag. 3) Options to add Google, Quantcast.com, GetClicky.com, Compete.com Analytics scripts for your blogs. Please go to `Settings -> All in One Webmaster` for more and detailed options. Version 7.0.4 | By Arpit Shah | Visit plugin site
☐ **AskApache Password Protect** Activate | Edit | Delete	Advanced Security: Password Protection, Anti-Spam, Anti-Exploits, more to come... Version 4.6.6 | By AskApache | Visit plugin site

3. Click on the **Settings** menu.
4. Click on the **Permalinks** option.
5. Click on the **Save Changes** button.
6. Click on the **WP Print Friendly** option in the **Settings** menu. You will see the **WP Print Friendly** page, as shown in the following screenshot:


WP Print Friendly Options ‹ WordPress 3 Cookbook — WordPress
WP Print Friendly Options ‹ Wor...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=wp_print_friendly
Google
admin
Add New
Comments
Appearance
Search
WordPress 3 Cookbook
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Reviews
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
WP Print Friendly
Save Changes
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

WP Print Friendly

Automatically add print links based on settings below? ● Yes ○ No

Automatically place link: ○ Above content ● Below content ○ Above and below content

Display automatically on: ☑ Posts ☑ Pages ☐ Reviews

Text for link to print entire item: Print this!

Text for link to print current page: Print this!

If viewing a multipage post (set by using the <!--nextpage--> tag), the text above is used for a link to print just the current page.

To hide this link, *clear the field's contents.*

CSS for print links: print_link

For page-specific print links, a second class, created by appending ***_cur*** *to the above text, is added to each link.*

Be aware that Internet Explorer will only interpret the first two CSS classes, so if multiple classes are entered above, the page-specific class may not be available in IE.

Open print-friendly views in: ○ In the same window ● In a new window

7. Select the options you want.
8. Click the **Save Changes** button.

How it works...

The **WP Print Friendly** plugin works by tapping into the templating system of WordPress 3. The plugin leverages features of the core, specifically the optional templates and the system CSS.

There's more...

To get the most out of this plugin, you need to do a bit more. You can style both the print link and the resulting print page.

To style the print link

As you can see in the screenshot above, the plugin specifies a class specifically for the purpose of handling the styling of the print link that appears on the pages and posts of your site. The default class is named `print_link`, but you can change it at any time from the **WP Print Friendly** settings page.

To take advantage of this feature, you will need to edit your theme's active stylesheet and add a definition for `.print_link`.

To style the print page

If you look inside the WP Print Friendly plugin directory you will find a file named `default-template.php`. To gain control over the formatting of your print friendly pages, follow these steps:

1. Copy the `default-template.php` file.
2. Paste the copy into your active theme's directory.
3. Rename the file `wpf.php`.
4. Open the file for editing.
5. Make any customizations you desire.
6. Save the file.

Now, when someone clicks on the print link on one of your articles, they will be shown the new template you just created. This feature makes it possible for you to tailor the print friendly pages to fit the identity of your site and to carry relevant information you want people to have when they print the article.

> Note that since the plugin uses the default templating functionality built into WordPress 3, you can also create template suggestions to be used in specific situations. If, for example, you want to have different print formats for various taxonomy terms, or for custom content types, you can do so. Follow the standard WordPress naming conventions to call the relevant template. To produce a template for use by a specific taxonomy term use, `wpf-[taxonomy-term].php`. To produce a template for use by a custom content type, use `wpf-[custom-content-type-name].php`.

Extending WordPress search

WordPress comes with a site search functionality. The default search is quite good, but if you want to obtain a more complete set of search results, you will need to look at installing a plugin that provides enhanced search capabilities.

In this recipe, we look at how you can index your site more completely and provide a more comprehensive set of search results for your site user.

Getting ready

To execute this recipe, you will need to install the **Search Engine** plugin. You will need to install this plugin before you can get started. Search for Search Engine inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://scottkclark.com/wordpress/search-engine/`

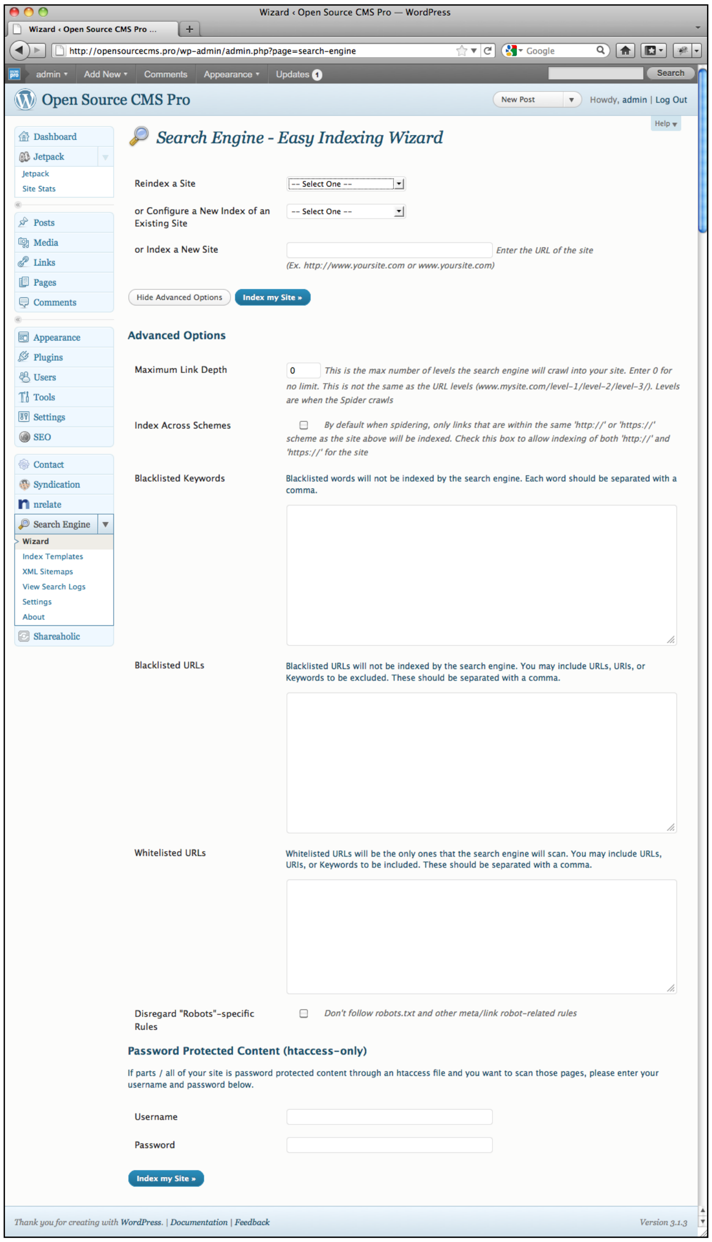
How to do it...

There's a bit of configuration needed for the plugin to work properly:

1. Log in to your WordPress **Dashboard**.
2. Click on **Search Engine** menu.

3. Click on **Wizard** option. The following screenshot will appear:

> Be patient. Site indexing can take a while!


Wizard ‹ Open Source CMS Pro — WordPress
Wizard ‹ Open Source CMS Pro ...
http://opensourcecms.pro/wp-admin/admin.php?page=search-engine
Google
admin
Add New
Comments
Appearance
Updates 1
Search
Open Source CMS Pro
New Post
Howdy, admin | Log Out
Help
Dashboard
Jetpack
Jetpack
Site Stats
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
SEO
Contact
Syndication
nrelate
Search Engine
Wizard
Index Templates
XML Sitemaps
View Search Logs
Settings
About
Shareaholic
Hide Advanced Options
Index my Site »
Index my Site »
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

Search Engine - Easy Indexing Wizard

Reindex a Site: -- Select One --

or Configure a New Index of an Existing Site: -- Select One --

or Index a New Site: *Enter the URL of the site*
(Ex. http://www.yoursite.com or www.yoursite.com)

Advanced Options

Maximum Link Depth: 0 *This is the max number of levels the search engine will crawl into your site. Enter 0 for no limit. This is not the same as the URL levels (www.mysite.com/level-1/level-2/level-3/). Levels are when the Spider crawls*

Index Across Schemes: ☐ *By default when spidering, only links that are within the same 'http://' or 'https://' scheme as the site above will be indexed. Check this box to allow indexing of both 'http://' and 'https://' for the site*

Blacklisted Keywords: Blacklisted words will not be indexed by the search engine. Each word should be separated with a comma.

Blacklisted URLs: Blacklisted URLs will not be indexed by the search engine. You may include URLs, URIs, or Keywords to be excluded. These should be separated with a comma.

Whitelisted URLs: Whitelisted URLs will be the only ones that the search engine will scan. You may include URLs, URIs, or Keywords to be included. These should be separated with a comma.

Disregard "Robots"-specific Rules: ☐ *Don't follow robots.txt and other meta/link robot-related rules*

Password Protected Content (htaccess-only)

If parts / all of your site is password protected content through an htaccess file and you want to scan those pages, please enter your username and password below.

Username

Password

4. Click **Show Advanced Options** to expose all the configuration options for the plugin.
5. Select your site's name from the **Configure a New Index of an Existing Site**. If your site name does not appear there, enter it in the field **Index a New Site**.
6. Select the configuration options you desire from the choices on the page.
7. Click on the **Index my Site** button.

How it works...

The **Search Engine** plugin indexes pages in a comprehensive fashion. Unlike the default WordPress search, which is focused exclusively on your article content, the **Search Engine** plugin indexes the pages as the visitors see them; that means not only posts and pages, but also custom content types, plugin generated content, sidebars, and widgets.

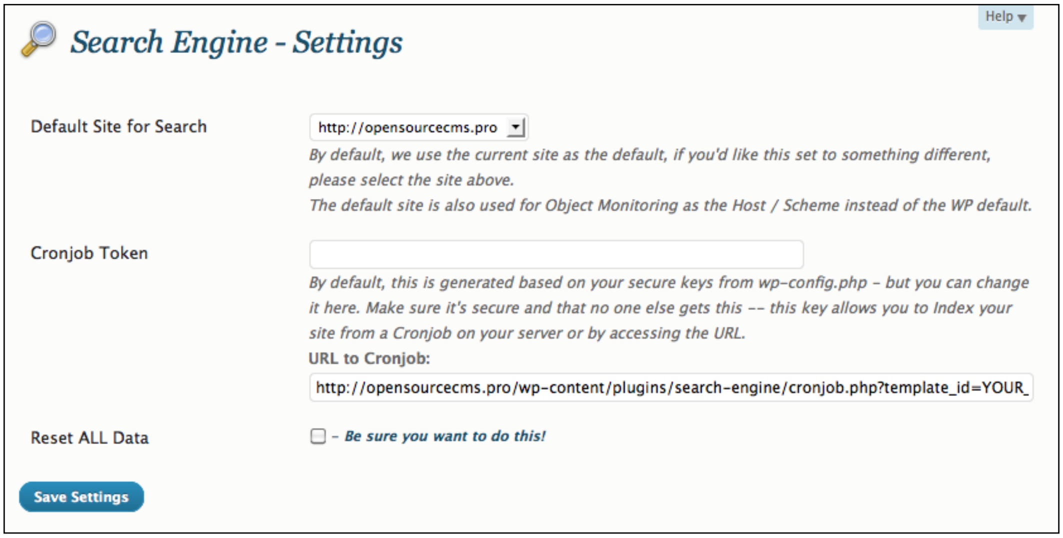
> You can control the indexing through the use of a `robots.txt` file. Files blocked in `robots.txt` will also not be indexed, unless you configure the plugin to ignore the `robots.txt` file. If you need more granular control, exclude content within a specific section of a page by added the class `.noindex` to the element tag.

There's more

You can index your site manually if you wish, but it's much more convenient to have it indexed frequently and automatically. To set up automatic indexing, you will need to have a server that supports cron. If you have access to cron, then follow these steps:

1. Log in to the WordPress **Dashboard**.
2. Click on the **Search Engine** menu.
3. Click on the **Settings** option.

4. Copy the information in the field **URL to Cronjob**, as you can see in the following screenshot:



Search Engine - Settings

Default Site for Search: http://opensourcecms.pro

By default, we use the current site as the default, if you'd like this set to something different, please select the site above.
The default site is also used for Object Monitoring as the Host / Scheme instead of the WP default.

Cronjob Token

By default, this is generated based on your secure keys from wp-config.php - but you can change it here. Make sure it's secure and that no one else gets this -- this key allows you to Index your site from a Cronjob on your server or by accessing the URL.

URL to Cronjob:

http://opensourcecms.pro/wp-content/plugins/search-engine/cronjob.php?template_id=YOUR_

Reset ALL Data ☐ - ***Be sure you want to do this!***

5. Access your web hosting control panel, or use SSH and work on the command line.
6. Access the cron functionality on the control panel.
7. Enter the data you copied from the URL to Cronjob.
8. Set the frequency you want the cron job to run.
9. Save the changes.

Your site will now be automatically indexed, at the frequency you have specified. To check to see when the last indexing was run, access the information in your Dashboard by clicking on the option **Index Templates** on the **Search Engine** menu.

Enhancing navigation with breadcrumbs

If you are looking for ways to improve your site's usability, adding a breadcrumb trail is definitely an option to consider.

> What's a breadcrumb trial? According to Wikipedia:
>
> Breadcrumbs typically appear horizontally across the top of a webpage, usually below any title bars or headers. They provide links back to each previous page that the user navigated through in order to get to the current page, for hierarchical structures usually the parent pages of the current one. Breadcrumbs provide a trail for the user to follow back to the starting/entry point of a website. Generally, a greater than symbol (>) is used as hierarchy separator, although other glyphs can be used to represent this.

In this recipe we explore using a plugin to add breadcrumbs functionality to your site.

Getting ready

To execute this recipe, you will need to install the Breadcrumbs Plus plugin. You will need to install this plugin before you can get started. Search for **Breadcrumbs Plus** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://snippets-tricks.org/proyecto/breadcrumbs-plus-plugin/`

In addition to installing the plugin, you will also need a code editor and access to the files of the WordPress installation on your server.

How to do it...

Setting up a breadcrumb trail involves two separate processes. First you must install the plugin, then you must add a line of code to your theme.

Once you have installed and activated the plugin, follow these steps to get the code into your theme files:

1. Access your WordPress installation on your server.
2. Download the template files you are using on your site.
3. Open each template file for editing.

4. Add to each file the following line of code:

   ```
   <?php breadcrumbs_plus(); ?>
   ```

   > Make sure you don't place the line of code inside of an existing php block!

5. Save the files and return them to your server, overwriting the original files.

If you visit your site, you should now see a breadcrumb trail on all the pages whose templates you modified.

> **Which templates to modify?**
>
> The answer depends on what theme you are using and which content types and site functions you have enabled. Looking at the default TwentyTen theme, most people would certainly want to add the breadcrumb function to the file `single.php`, which controls the display of single posts and `page.php`, which controls the display of the page content type. Additionally, if you are using categories, you will want to modify `category.php`. If you are using tags, also modify `tags.php`. If you are using the archive functionality, modify `archive.php`. Another option is to add breadcrumbs to the author pages, by modifying `author.php`. If you have created custom content types, or you are using a more complex theme, this list may well be longer.

How it works...

The Breadcrumbs Plus plugin enables the `breadcrumbs_plus ()` function. The function includes a number of variables you can use to customize your breadcrumb display, as discussed in the next section.

You control the placement of the breadcrumbs on the page by where you place the function inside the template files, and through the use of CSS styling.

There's more...

The Breadcrumbs Plus plugin includes a set of arguments that can be added to the function to customize the display. The function syntax is as follows:

```
<?php breadcrumbs_plus( $args ); ?>
```

The possible arguments include:

- `prefix`: Used to add things, typically code for styling, before the breadcrumbs.
- `suffix`: Used to add things, typically the closing tags for styling, after the breadcrumbs.

- `title`: The text shown before the first link in the breadcrumb trail.
- `home`: The name to be shown for the home page link.
- `sep`: The character to be used in between each item in the breadcrumb trail.
- `front_page`: Set to true to show the breadcrumb trail on the home page.
- `bold`: Set to true to make the text in the breadcrumb trail bold.
- `show_blog`: Set to true to display the name of the blog before individual posts.
- `echo`: The echo parameter appears to have no functionality in the present version of the plugin.
- `singular_post_taxonomy`: Set to display the category to which a single post belongs.

The default values are:

- `'prefix' => '<p>'`
- `'suffix' => '</p>'`
- `'title' => _( 'You are here', 'breadcrumbs-plus' )`
- `'home' => _( 'Home', 'breadcrumbs-plus' )`
- `'sep' => '>>',`
- `'front_page' => false`
- `'bold' => true`
- `'show_blog' => true`
- `'echo' => true`
- `'singular_post_taxonomy' => 'category'`

Let's look at an example of how to use this information. The default formatting provides output like this:

You are here: Home >> [category] >> [post name]

We want to construct a custom breadcrumb trail that looks like this:

History : Top > [category] > [post title]

To achieve this change in the output, we need to modify the function to override the default values. The customized function would look like this:

```
<?php breadcrumbs_plus( array( 'title' => 'History : ', 'home' =>
     'Top', 'sep' => '>') ); ?>
```

> You can also modify the styling of the breadcrumbs trail by adding the class `.breadcrumbs-plus` to your active theme's stylesheet.

Stopping SPAM

All website owners have to be alert for SPAM postings on their websites. Monitoring your site and policing comments to remove unwanted commercial solicitations is not only time consuming, but also prone to error. SPAM can clog up the comments on your site and create a bad impression of your professionalism. Moreover, it can also be a potential security issue, as SPAM can be used as a vector for malware.

In this recipe, we look at implementing the Akismet system to help automatically monitor your site for SPAM comments and keep them from appearing to your site visitors.

Getting ready

To execute this recipe, you will be using the Akismet plugin. This plugin is included in the WordPress core. You should not need to install it – unless you have previously deleted it from your system. Check the **Plugins** page to see if it is there and if so, activate it. If it is not present, you will need to install this plugin before you can get started. Search for **Akismet** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

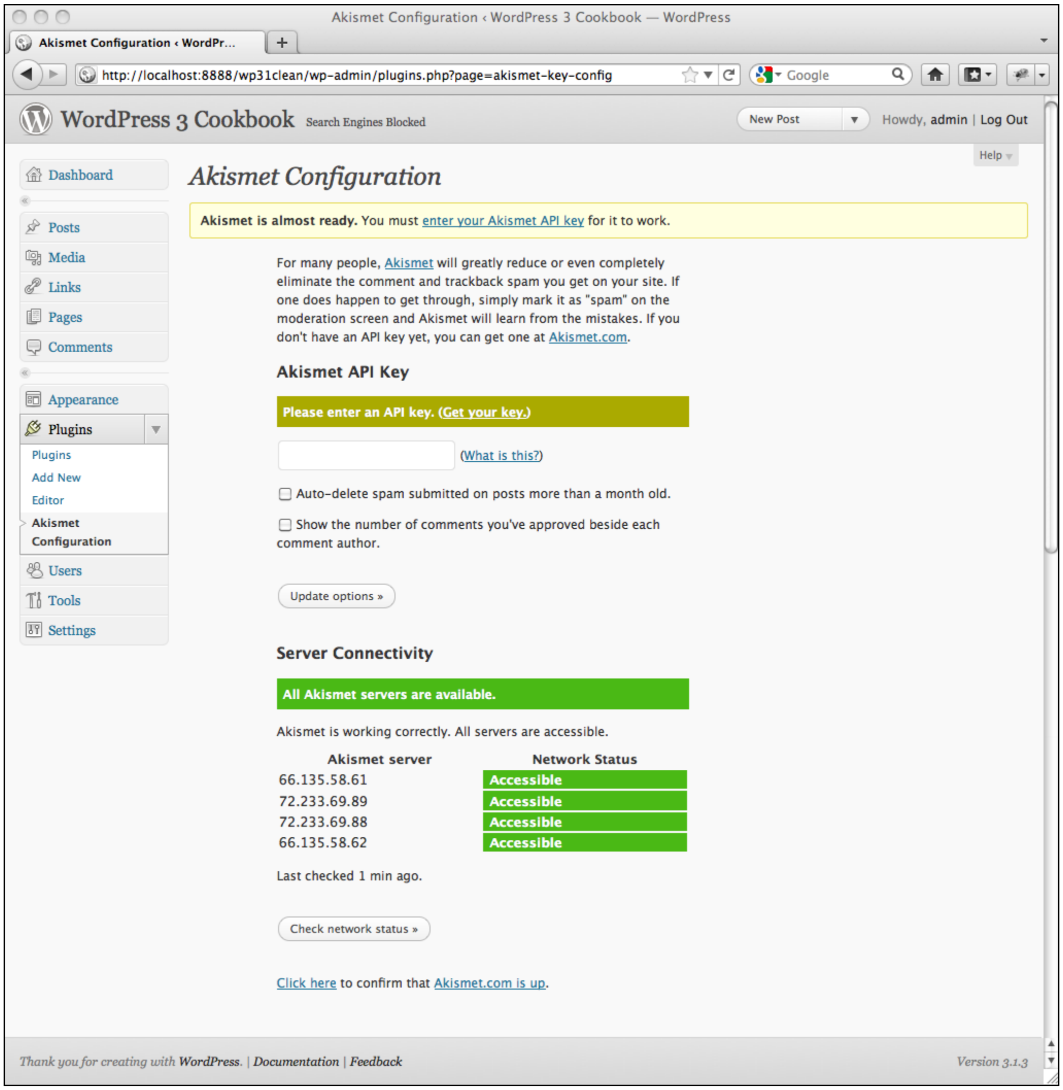
> You can learn more about the plugin by visiting the developer's website at `http://akismet.com/`

How to do it...

To create featured posts, simply add it to the "featured" category. The five most recent posts will be shown on your blog homepage.

1. Log in to your WordPress **Dashboard**.
2. Click on the **Plugins** menu.
3. Click on the **Plugins** option.
4. If you have not done so already, click on the **Activate** link for the **Akismet** plugin.
5. The system will provide a notification that you need to enter an **Akismet API key**. Either click the link in the warning, or click the option **Akismet Configuration** on the **Plugin** menu.

6. On the **Akismet Configuration** page, shown in the next screenshot, you will need to enter your **Akismet API key**. If you do not have a key, you will need to visit `http://akismet.com` and obtain a key. Keys are free for personal sites.
7. Select any other options you desire.
8. Click on the **Update options** button.


Akismet Configuration ‹ WordPress 3 Cookbook — WordPress
Akismet Configuration ‹ WordPr...
http://localhost:8888/wp31clean/wp-admin/plugins.php?page=akismet-key-config
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Akismet Configuration
Users
Tools
Settings

Akismet Configuration

Akismet is almost ready. You must enter your Akismet API key for it to work.

For many people, Akismet will greatly reduce or even completely eliminate the comment and trackback spam you get on your site. If one does happen to get through, simply mark it as "spam" on the moderation screen and Akismet will learn from the mistakes. If you don't have an API key yet, you can get one at Akismet.com.

Akismet API Key

Please enter an API key. (Get your key.)

(What is this?)

☐ Auto-delete spam submitted on posts more than a month old.

☐ Show the number of comments you've approved beside each comment author.

Update options »

Server Connectivity

All Akismet servers are available.

Akismet is working correctly. All servers are accessible.

Akismet server	Network Status
66.135.58.61	Accessible
72.233.69.89	Accessible
72.233.69.88	Accessible
66.135.58.62	Accessible

Last checked 1 min ago.

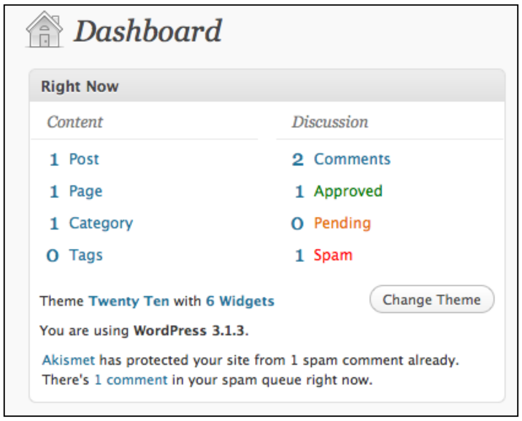
Check network status »

Click here to confirm that Akismet.com is up.

Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

How it works...

Akismet was created by Automattic, the creators of WordPress. The plugin is designed to provide a system for reporting and then identifying SPAM and SPAMmers. When a comment is posted, Akismet will assess it on a variety of factors in an attempt to identify whether it is potential SPAM. If Akismet suspects it is SPAM, the comment will automatically be put into the SPAM queue and a notification will appear on your dashboard, as seen in the following screenshot :



Content	Discussion
1 Post	2 Comments
1 Page	1 Approved
1 Category	0 Pending
0 Tags	1 Spam

Theme Twenty Ten with 6 Widgets

You are using WordPress 3.1.3.

Akismet has protected your site from 1 spam comment already.
There's 1 comment in your spam queue right now.

You can then check the comment in the queue. If it is SPAM you can delete it permanently. If it is not SPAM, you can mark it as **Not Spam** and return it to the publication queue. Over time, the Akismet system will continue to refine its SPAM detection abilities and learn from what you tell it about the content of the comments on the site.

Optimizing performance with cache management

As both website users and website owners, we want sites to load quickly without long delay. With Google now looking at site performance as a factor in search rankings, it is even more important that a site performs well relative to competitors. Site performance is impacted by a number of factors; one of the key issues is the amount of time it takes the server to generate pages on the site. An easy way to improve your site's performance is to implement caching, which takes some of the load off the server by making copies of frequently used files and displaying them to site visitors, instead of relying on the server to construct/retrieve the files every single time they are requested.

In this recipe, we look at implementing the powerful **WP Super Cache** plugin. When properly configured, this plugin can achieve considerable improvements in site performance.

Getting ready

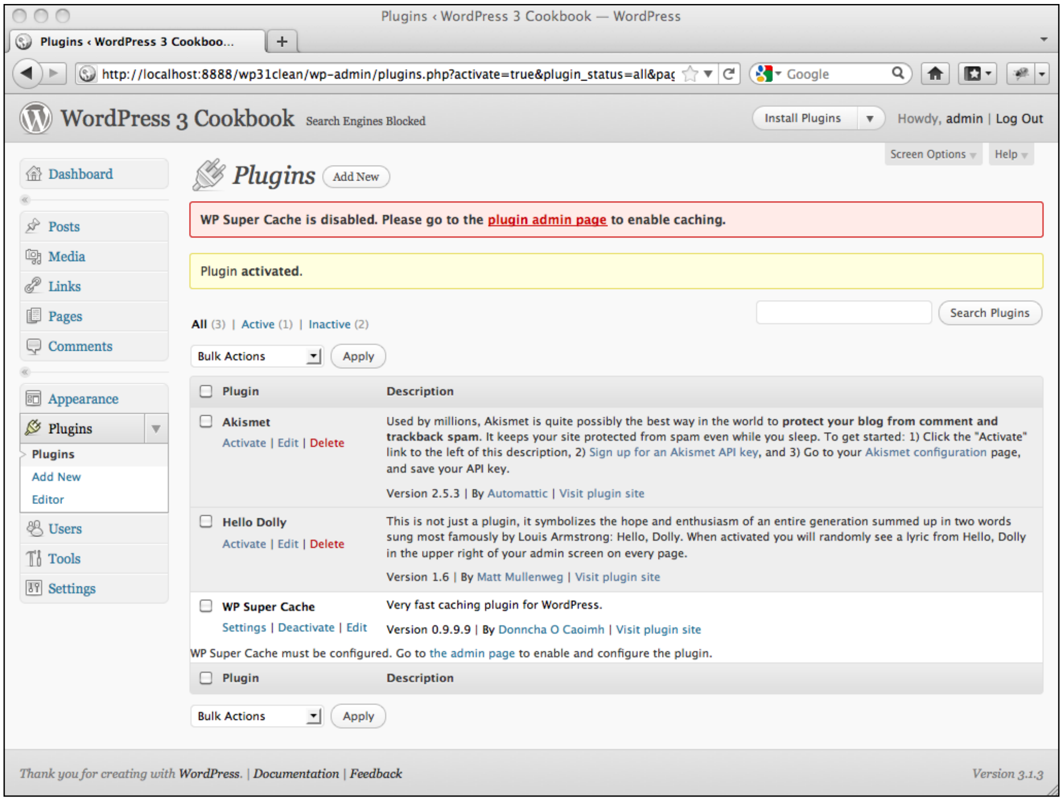
To execute this recipe, you will need to install the **WP Super Cache** plugin. You will need to install this plugin before you can get started. Search for **WP Super Cache** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://ocaoimh.ie/wp-super-cache/`

How to do it...

The **WP Super Cache** plugin is a complex piece of code with a number of configuration options. Let's look at basic setup first:

1. After you install and activate the plugin, you will see a notification that the plugin must be enabled, as shown in the following screenshot:



Plugins ‹ WordPress 3 Cookbook — WordPress
Plugins ‹ WordPress 3 Cookboo...
http://localhost:8888/wp31clean/wp-admin/plugins.php?activate=true&plugin_status=all&pag
Google
WordPress 3 Cookbook
Search Engines Blocked
Install Plugins
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Settings
Plugins
Add New
All (3) | Active (1) | Inactive (2)
Search Plugins
Bulk Actions
Apply
Plugin
Description
Akismet
Activate | Edit | Delete
Hello Dolly
Activate | Edit | Delete
WP Super Cache
Settings | Deactivate | Edit
Plugin
Description
Bulk Actions
Apply
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

WP Super Cache is disabled. Please go to the plugin admin page to enable caching.

Plugin activated.

Akismet: Used by millions, Akismet is quite possibly the best way in the world to **protect your blog from comment and trackback spam**. It keeps your site protected from spam even while you sleep. To get started: 1) Click the "Activate" link to the left of this description, 2) Sign up for an Akismet API key, and 3) Go to your Akismet configuration page, and save your API key.

Version 2.5.3 | By Automattic | Visit plugin site

Hello Dolly: This is not just a plugin, it symbolizes the hope and enthusiasm of an entire generation summed up in two words sung most famously by Louis Armstrong: Hello, Dolly. When activated you will randomly see a lyric from Hello, Dolly in the upper right of your admin screen on every page.

Version 1.6 | By Matt Mullenweg | Visit plugin site

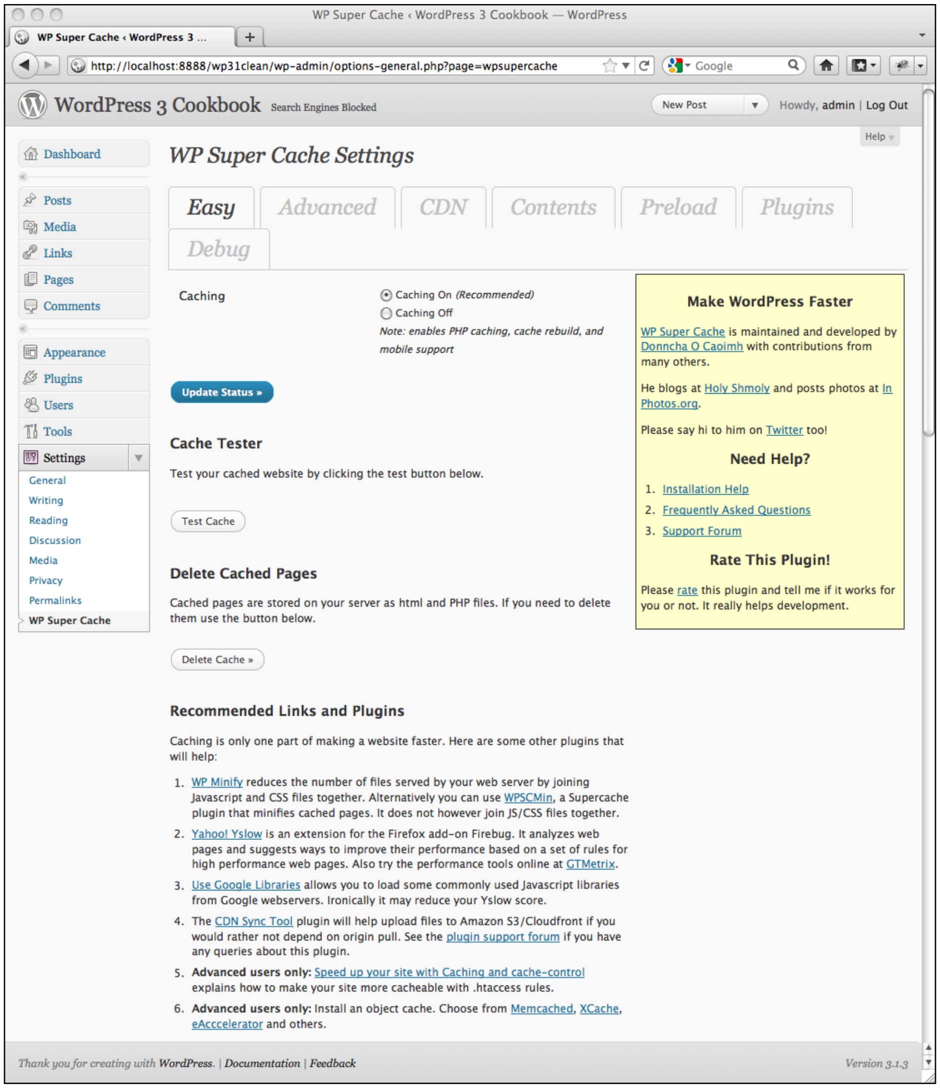
WP Super Cache: Very fast caching plugin for WordPress.

Version 0.9.9.9 | By Donncha O Caoimh | Visit plugin site

WP Super Cache must be configured. Go to the admin page to enable and configure the plugin.

2. Click on the **Settings** menu.
3. Click on the option **WP Super Cache**.
4. On the **WP Super Cache Settings** page, seen in the next screenshot, select the choice **Caching On**.
5. Click on **Update Status**.

Caching is now enabled for your site; however, to get the most out of the plugin, you need to explore the options found on the various tabs seen at the top of the **WP Super Cache Settings** page, as shown in the following screenshot:


WP Super Cache ‹ WordPress 3 Cookbook — WordPress
WP Super Cache ‹ WordPress 3 ...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=wpsupercache
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
WP Super Cache
WP Super Cache Settings
Easy
Advanced
CDN
Contents
Preload
Plugins
Debug
Update Status »
Test Cache
Delete Cache »
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

Caching

● Caching On *(Recommended)*
○ Caching Off

Note: enables PHP caching, cache rebuild, and mobile support

Cache Tester

Test your cached website by clicking the test button below.

Delete Cached Pages

Cached pages are stored on your server as html and PHP files. If you need to delete them use the button below.

Recommended Links and Plugins

Caching is only one part of making a website faster. Here are some other plugins that will help:

1. WP Minify reduces the number of files served by your web server by joining Javascript and CSS files together. Alternatively you can use WPSCMin, a Supercache plugin that minifies cached pages. It does not however join JS/CSS files together.
2. Yahoo! Yslow is an extension for the Firefox add-on Firebug. It analyzes web pages and suggests ways to improve their performance based on a set of rules for high performance web pages. Also try the performance tools online at GTMetrix.
3. Use Google Libraries allows you to load some commonly used Javascript libraries from Google webservers. Ironically it may reduce your Yslow score.
4. The CDN Sync Tool plugin will help upload files to Amazon S3/Cloudfront if you would rather not depend on origin pull. See the plugin support forum if you have any queries about this plugin.
5. **Advanced users only:** Speed up your site with Caching and cache-control explains how to make your site more cacheable with .htaccess rules.
6. **Advanced users only:** Install an object cache. Choose from Memcached, XCache, eAcccelerator and others.

Make WordPress Faster

WP Super Cache is maintained and developed by Donncha O Caoimh with contributions from many others.

He blogs at Holy Shmoly and posts photos at In Photos.org.

Please say hi to him on Twitter too!

Need Help?

1. Installation Help
2. Frequently Asked Questions
3. Support Forum

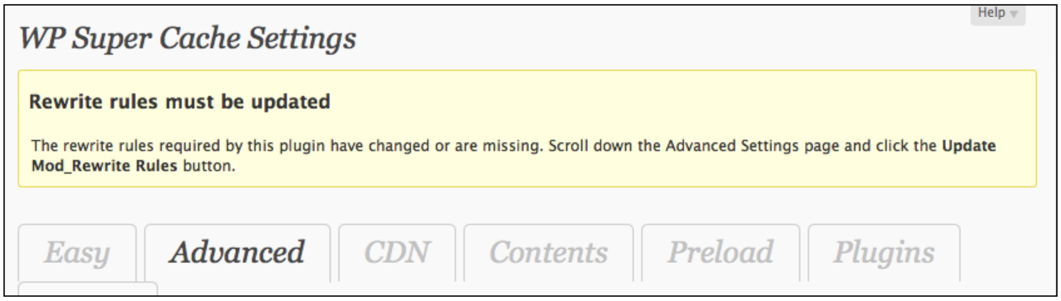
Rate This Plugin!

Please rate this plugin and tell me if it works for you or not. It really helps development.

Follow these steps to configure the plugin:

1. On the **WP Super Cache Settings** page, select the tab **Advanced**.
2. Select the option **Cache hits to this website for quick access**.
3. Select the option **Use mod_rewrite to serve cache files**.
4. Select **Compress pages so they're served more quickly to visitors**.
5. Select **304 Not Modified browser caching. Indicate when a page has not been modified since last requested**.
6. Select **Don't cache pages for known users**.
7. Select **Cache rebuild. Serve a supercache file to anonymous users while a new file is being generated**.
8. Select **Extra homepage checks**.
9. Click **Update Status**.

The system will display a notice that the **Rewrite rules must be updated**, as shown in the following screenshot:



WP Super Cache Settings

Rewrite rules must be updated

The rewrite rules required by this plugin have changed or are missing. Scroll down the Advanced Settings page and click the **Update Mod_Rewrite Rules** button.

Scroll down the page and click on the button labeled **Update Mod_Rewrite Rules**.

> You probably also want to exclude certain types of pages from being cached, particularly search results and feeds. You can select those options, and others, on the **Advanced** tab.

How it works...

WP Super Cache's purpose is static caching. The plugin generates HTML files that are served directly by Apache without processing comparatively heavy PHP scripts, resulting in a significant increase in speed for serving of your WordPress blog.

There's more...

This plugin supports a number of options. You should explore all the tabs to learn what choices appear. As you grow more familiar with the plugin, you may wish to be more aggressive about your caching. Let's look at two techniques you can apply.

Using a content delivery network

WP Super Cache offers CDN support. CDN is a content delivery network, which is a system of computers that contain cached copies of data. As the network distributes the load between multiple computers, it tends to increase performance and reduce latency. By default, the system uses origin pull to populate the CDN. If you prefer to upload and synch the files yourself, you can use the **CDN Sync Tool** plugin.

> You can set up CDN support by clicking on the CDN tab on the WP Super Cache Settings page.

Preloading content

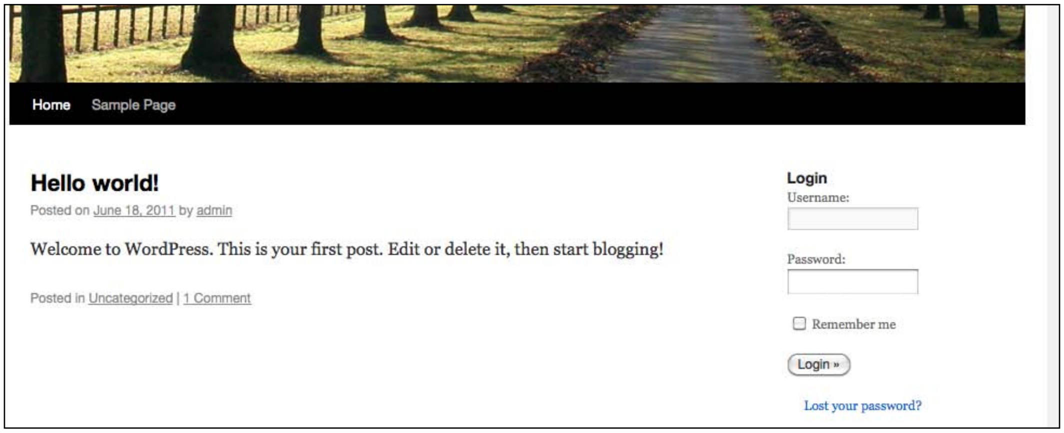
Another technique that can improve performance and can also help your Google ranking, is the selective use of preloading. Click on the **Preload** tab on the **WP Super Cache Settings** page to see the options available. Select the option **Preload mode**, then click on **Preload Cache Now**. The system will make static copies of the pages on your website and serve them to visitors.

> If you have a very large site, you should limit the number of items preloaded, as the plugin will create a large number of files to support the caching. If you are on a shared webhost, excessive preloading may cause you problems, or may even be prohibited by the host.

Displaying a login form

Having a login form visible on the page will improve the usability of your site. Instead of having to click on a link and wait for a new page to load, users can directly enter their username and password and immediately log in.

In this recipe, we implement the Sidebar Login plugin to add not only a login form, but also password reminder and registration links. The following screenshot shows the plugin in action, with the login widget assigned to the right sidebar of the default theme:



Hello world!

Posted on June 18, 2011 by admin

Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!

Posted in Uncategorized | 1 Comment

Getting ready

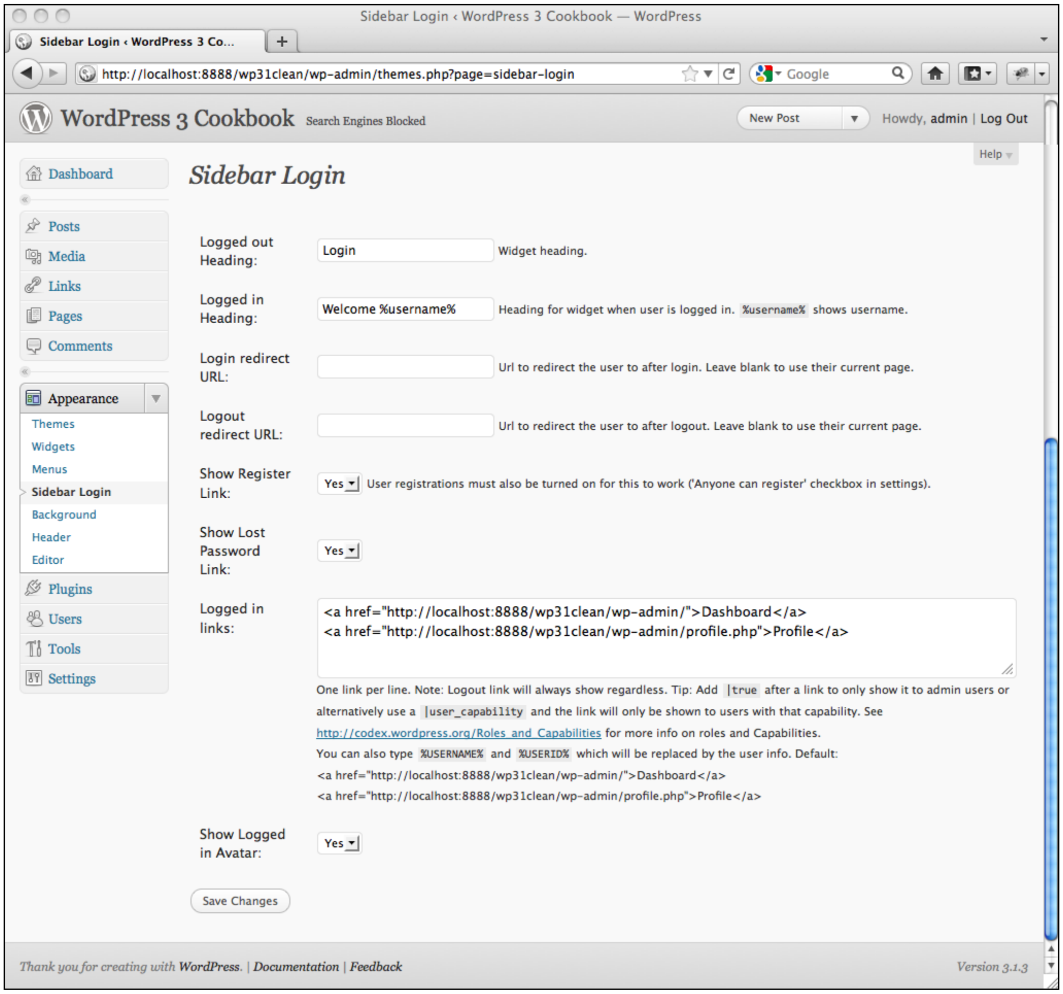
To execute this recipe, you will need to install the **Sidebar Login** plugin. You will need to install this plugin before you can get started. Search for **Sidebar Login** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://wordpress.org/extend/plugins/sidebar-login/`

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.
3. Click on the option **Sidebar Login**.
4. On the **Sidebar Login** configuration screen, shown in the next screenshot, you can customize the appearance of the login box and related links.
5. Click on the **Save Changes** button.
6. Click on the **Widgets** link on the **Appearance** menu.
7. Grab the **Sidebar Login** widget and drag it to the widget area where you want it to appear.

The widget will now appear on your pages, in the widget area you assigned it to.


Sidebar Login ‹ WordPress 3 Cookbook — WordPress
Sidebar Login ‹ WordPress 3 Co...
http://localhost:8888/wp31clean/wp-admin/themes.php?page=sidebar-login
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Themes
Widgets
Menus
Sidebar Login
Background
Header
Editor
Plugins
Users
Tools
Settings
Save Changes
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

Sidebar Login

Logged out Heading: Login — Widget heading.

Logged in Heading: Welcome %username% — Heading for widget when user is logged in. `%username%` shows username.

Login redirect URL: — Url to redirect the user to after login. Leave blank to use their current page.

Logout redirect URL: — Url to redirect the user to after logout. Leave blank to use their current page.

Show Register Link: Yes — User registrations must also be turned on for this to work ('Anyone can register' checkbox in settings).

Show Lost Password Link: Yes

Logged in links:

```
<a href="http://localhost:8888/wp31clean/wp-admin/">Dashboard</a>
<a href="http://localhost:8888/wp31clean/wp-admin/profile.php">Profile</a>
```

One link per line. Note: Logout link will always show regardless. Tip: Add `|true` after a link to only show it to admin users or alternatively use a `|user_capability` and the link will only be shown to users with that capability. See http://codex.wordpress.org/Roles_and_Capabilities for more info on roles and Capabilities.
You can also type `%USERNAME%` and `%USERID%` which will be replaced by the user info. Default:
Dashboard
Profile

Show Logged in Avatar: Yes

How it works...

The plugin simply provides an alternative to the default login form on the dedicated login page. In addition to the standard functionality, the plugin also allows you to control redirects after login and provides useful links once the user is authenticated.

Displaying related posts

As a site owner, you want to give visitors reasons to stay on your site. One of the most effective methods for increasing page views is to display a list of related posts. Providing the visitors with a list of other content items that are similar to the one they are viewing is not only a good deal for site owners, but also for visitors, as it provides them with access to richer information about a topic.

In this recipe, we look at implementing a plugin to add related posts functionality to your site. In the first part of the recipe, we look at a basic plugin. In the second part of the recipe we look at an alternative plugin solution.

Getting ready

To execute this recipe, you will need to install the All Related Posts plugin. You will need to install this plugin before you can get started. Search for **All Related Posts** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

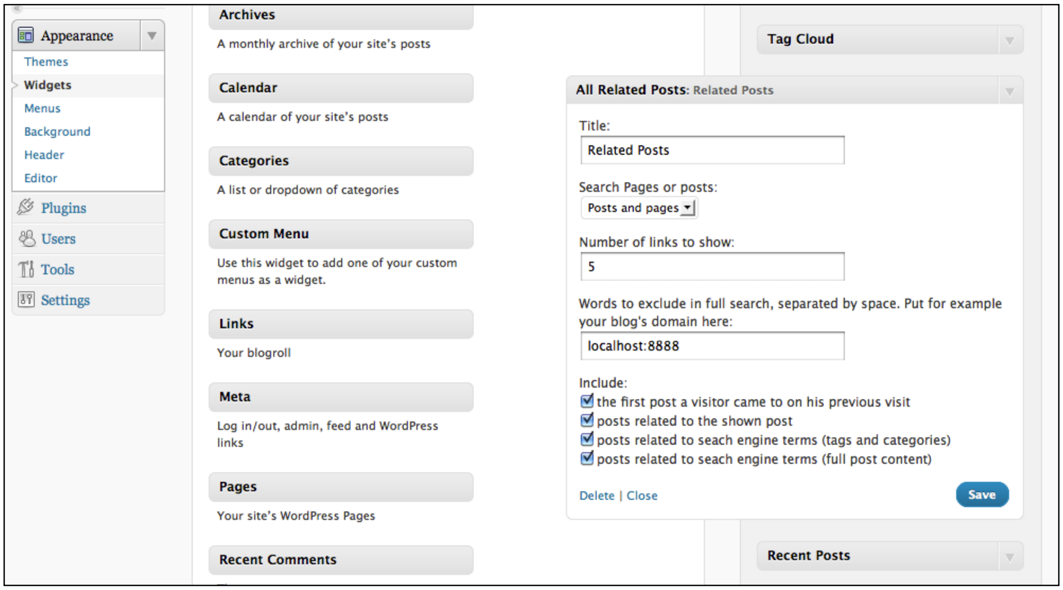
> You can learn more about the plugin by visiting the developer's website at `http://blog.bigcircle.nl/about/wordpress-plugins/`

How to do it...

To configure this plugin and enable it on the site, follow these simple steps:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.
3. Click on the **Widgets** option.
4. Find the **All Related Posts** widget and drag it to one of your site's Primary Widget Areas.

5. Click on the arrow on the widget's title bar to expand it and view the configuration options, as shown in the following screenshot:


Appearance
Themes
Widgets
Menus
Background
Header
Editor
Plugins
Users
Tools
Settings
Archives
A monthly archive of your site's posts
Calendar
A calendar of your site's posts
Categories
A list or dropdown of categories
Custom Menu
Use this widget to add one of your custom menus as a widget.
Links
Your blogroll
Meta
Log in/out, admin, feed and WordPress links
Pages
Your site's WordPress Pages
Recent Comments
Tag Cloud
Recent Posts
Delete | Close
Save

All Related Posts: Related Posts

Title: Related Posts

Search Pages or posts: Posts and pages

Number of links to show: 5

Words to exclude in full search, separated by space. Put for example your blog's domain here: localhost:8888

Include:
☑ the first post a visitor came to on his previous visit
☑ posts related to the shown post
☑ posts related to seach engine terms (tags and categories)
☑ posts related to seach engine terms (full post content)

6. Select the options you wish to use.
7. Click on the **Save** button.

If you now visit the front-end of your WordPress site, you should see the plugin in action. If it is not showing any links inside the related posts widget, click through a few pages; you should start to see output in the widget.

How it works...

The **All Related Posts** plugin uses a variety of methods for selecting the posts that are displayed. You can see the choices available in the previous screenshot:

- **the first post a visitor came to on his previous visit**
- **posts related to the shown post**
- **posts related to search engine terms (tags and categories)**
- **posts related to search engine terms (full post content)**

Of the choices, the second is the most commonly used criteria for determining whether posts are related. The other options offered by the plugin, however, are interesting. The first option relates to repeat visitors; if the visitor has been to the site in the past, the plugin will show them things they looked at on the previous visit. The latter two choices both affect visitors who come to your site through a search engine; the posts shown in the widget will be based on the words that were entered in the search query.

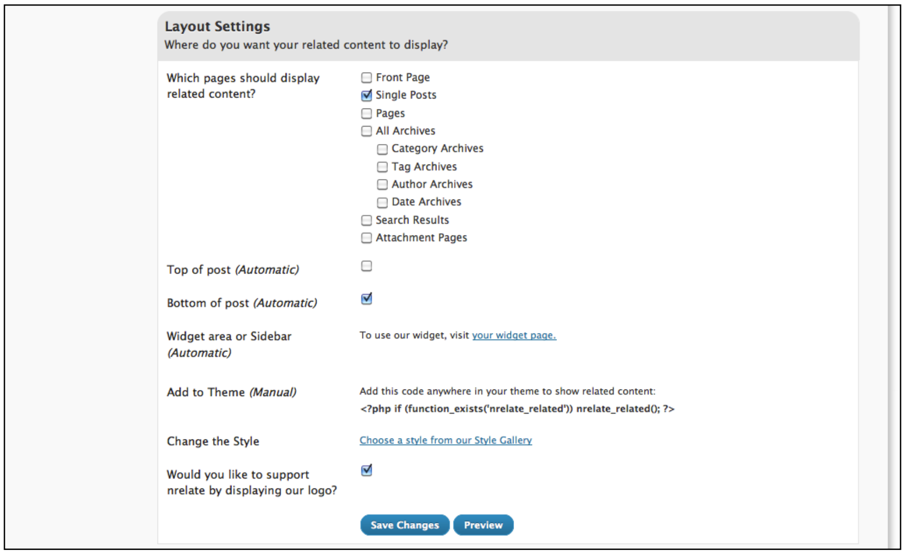
You can use any, or all of the options. The best approach is to set one up and view the results.

There's more...

While the **All Related Posts** plugin is a good solution, if you want more control, and more options, consider the nrelate Related Content plugin. The plugin works in a completely different fashion. With nrelate, your site will be indexed by nrelate's own spider. The indexing will create a listing of all the content on your site and thereafter, the plugin will draw up the index in order to find the most relevant onsite content.

Though nrelate offers only one matching methodology for identifying related content, it is extremely good at producing posts on the basis of relevance. You can configure both the degree of relevance and the depth of the indexing, giving you the option to provide highly relevant recommendations to your site visitors.

Another attractive feature of this plugin is the flexibility in layout. As you can see in the next screenshot, you have multiple choices for placing the related content list; you are not restricted to widget areas (though the plugin does also include a widget). If you prefer, you can also embed the output directly into your theme files. nrelate also comes with some styling options or, if you prefer, you can remove all styling and use the default theme styles.



Layout Settings
Where do you want your related content to display?

Which pages should display related content?	☐ Front Page ☑ Single Posts ☐ Pages ☐ All Archives ☐ Category Archives ☐ Tag Archives ☐ Author Archives ☐ Date Archives ☐ Search Results ☐ Attachment Pages
Top of post *(Automatic)*	☐
Bottom of post *(Automatic)*	☑
Widget area or Sidebar *(Automatic)*	To use our widget, visit your widget page.
Add to Theme *(Manual)*	Add this code anywhere in your theme to show related content: `<?php if (function_exists('nrelate_related')) nrelate_related(); ?>`
Change the Style	Choose a style from our Style Gallery
Would you like to support nrelate by displaying our logo?	☑

Creating a Featured Posts block

In the default WordPress installation, new posts on your site are displayed on the home page in chronological order. While you can control the output on the home page to a certain extent, what you cannot do is select specific articles to be featured at the top of the page, or in a widget area.

In this recipe we look at a solution that allows you to mark posts you want to be featured and then display those together as a group. This recipe relies on a plugin and modification to your theme files.

Getting ready

To execute this recipe, you will need to install the **List Category Posts** plugin. You will need to install this plugin before you can get started. Search for **All Related Posts** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

In addition, you will also need a code editor and access to your WordPress installation on your server.

> You can learn more about the plugin by visiting the developer's website at `http://foro.picandocodigo.net/categories/list-category-posts`

How to do it...

There are two parts to this recipe. In the first, we create a dedicated category to use for our featured posts; in the second, we modify the theme files to display the category. Let's start with the creation of a new category:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Posts** menu.
3. Click on the option **Categories**.
4. In the **Name field**, enter **Featured**.
5. Click on the **Add New Category** button.
6. The newly-created **Featured** category will now appear in the list of categories on the right side of the page. Click on the **edit** button, immediately below the Featured category's name.

7. The edit category screen will load. Look at the URL in the browser's address bar and find the value for **category&tag_ID**. Note that value as you will need it in the next part of this recipe.

To make this feature work, you will need to have at least one post assigned to the new Featured category, so assign a couple for testing purposes and let's move on to the next part of the recipe. Next, let's open the theme files and add the shortcode enabled by the plugin:

1. Access your WordPress installation on the server.
2. We're going to add the **Featured Posts** list to the top of the home page only, so open the `index.php` file for editing.
3. Add the following code where you want the **Featured Posts** block to appear:

```
<h2>Featured Posts</h2>
<?php echo do_shortcode("[catlist id=9 excerpt=yes]"); ?>
<hr />
```

4. Save the file.

Now, if you view the home page of your site, you should see at the top of the content area a list of all posts assigned to the Featured category.

> You could also place this block of posts inside a content item, but the syntax for the use of the shortcode will be different. Instead of wrapping the shortcode in PHP tags, you would simply switch your editor into HTML mode and type the following into the body of the post: `[catlist id=9 excerpt=yes]`

How it works...

The **List Category Posts** plugin enables for your use a new shortcode, `catlist id=x`, where x is the ID of the category you wish to display. Shortcodes can be used inside content items, or inside widgets, or inside your templates. In this recipe, we want to display Featured Posts at the top of our home page, so we edited the template being used for the home page in this theme.

> This recipe uses the default TwentyTen theme. If your theme uses a different file for the home page, for example, `home.php`, then you will modify that file, not `index.php`. Note that this will impact where the output appears. If, for example, your theme uses the `index.php` file for multiple pages, your output will show on all.

There's more...

This plugin enables a number of parameters that allow you to customize the output of the shortcode. In the example code above you see one of those parameters being used, `excerpt`.

The available parameters are shown in the following table:

Parameters	**Purpose**	**Valid values**
`id`	Specify the category to display, by category ID. You can display multiple categories by separating them with a for example, `id=1,3,5`	Integer
`name`	Specify the category to display, by category name.	Text
`tags`	Display posts from a specific tag.	Text
`orderby`	Customize the order.	`author`, `category`, `content`, `date`, `ID`, `menu_order`, `mime_type`, `modified`, `name`, `parent`, `password`, `rand`, `status`, `title`, `type`.
`order`	Set alphabetical order.	`ASC`, `DESC`
`numberposts`	Number of posts to display. Default is 5. Set to -1 to remove limitation.	Integer
`date`	Display the post date. The default is `no`.	`yes`, `no`
`author`	Display the post author's name. The default is `no`.	`yes`, `no`
`dateformat`	Format of the date output.	See, `http://codex.wordpress.org/Formatting_Date_and_Time` for possible formats
`template`	Used to specify a template for the output.	Template name, without the file extension.
`excerpt`	Include an excerpt from the post. Default is `no`.	`yes`, `no`
`excludeposts`	IDs of posts to exclude from the list.	Integer
`offset`	Pass over one or more initial posts.	Integer

Parameters	Purpose	Valid values
`content`	Show the full content of the post. Default is `no`.	`yes`, `no`
`catlink`	Show title of category with a link to the category. Default is `no`.	`yes`, `no`
`comments`	Show comment count for each post. Default is `no`.	`yes`, `no`
`thumbnails`	Show post thumbnails. Default is `no`.	`yes`, `no`
`post_type`	The type of post to show. Default is `post`.	`post`, `page`, `attachment`, `any`
`post_parent`	Show only the children of the post with this ID.	Integer
`class`	CSS class for the UL generated.	Text

Note that you can also create custom fields. To do so, you must specify two values: `customfield_name` and `customfield_value`. Both values must be defined for the custom field to work.

Adding a sitemap for your site visitors

Adding a sitemap to the front-end of your site has two key advantages: First, a sitemap can improve usability by helping visitors find content on your site and navigate more easily. Second, a sitemap can deliver search engine ranking benefits as it increases internal link density and can also promote fuller indexing of your site's contents.

In this recipe we look at implementing a plugin to automatically generate and maintain a sitemap for the front-end of your website.

> This recipe is not about building XML sitemaps; those are intended for the search engines, not for your site visitors. The topic of XML sitemaps is dealt with in *Chapter 7*, *Making an SEO Friendly Site*.

Getting ready

To execute this recipe, you will need to install the **HTML Sitemap Generator** plugin. You will need to install this plugin before you can get started. Search for **Html Sitemap Generator** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://cranesandskyhooks.com/wordpress-plugins/html-sitemap-generator/`

How to do it...

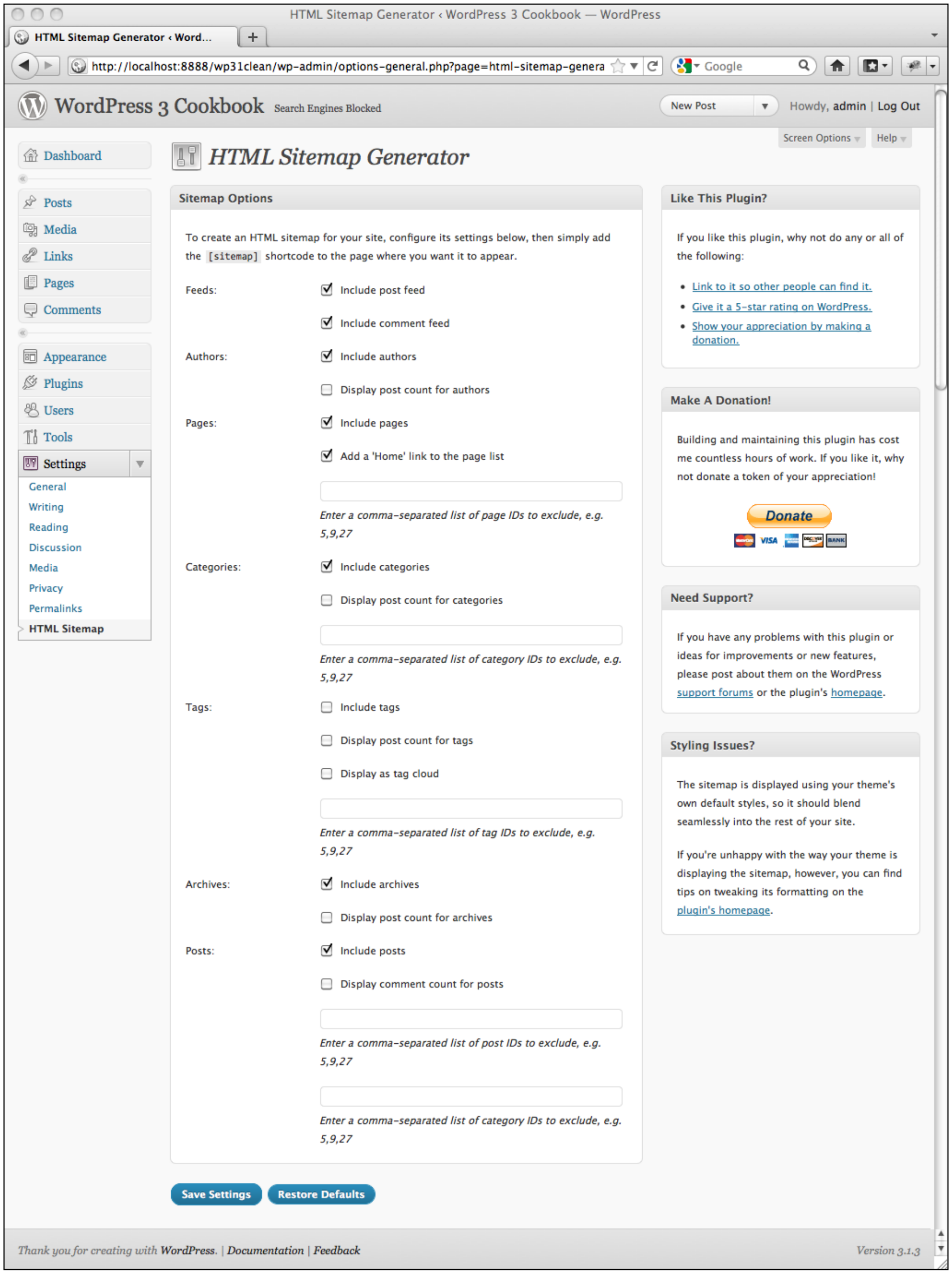
There are two parts to this recipe. You must configure the sitemap settings and you must add the shortcode to a page on your site to display the sitemap:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Settings** menu.
3. Click on the option **HTML Sitemap**.
4. View the options and select those you desire for your site. The options are shown in the next screenshot.
5. Click on the **Save Settings** button.

Next, let's set up a dedicated page to hold the sitemap:

1. Click the **Pages** menu.
2. Click on the option **Add New**.
3. In the **Title** field, enter **Site Map**.
4. In the editor box, switch to **HTML**.
5. Enter the sitemap shortcode as follows: `[sitemap]`
6. Click on the **Publish** button.

In the default installation with the TwentyTen theme installed, the steps above are all you need to do. With another theme, however, you may need to add the new page to one of your theme's menus to make it visible to site visitors.


HTML Sitemap Generator ‹ WordPress 3 Cookbook — WordPress
HTML Sitemap Generator ‹ Word...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=html-sitemap-genera
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Screen Options
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
HTML Sitemap
Donate
Save Settings
Restore Defaults
Thank you for creating with WordPress. | Documentation | Feedback
Version 3.1.3

HTML Sitemap Generator

Sitemap Options

To create an HTML sitemap for your site, configure its settings below, then simply add the `[sitemap]` shortcode to the page where you want it to appear.

Feeds: ☑ Include post feed

☑ Include comment feed

Authors: ☑ Include authors

☐ Display post count for authors

Pages: ☑ Include pages

☑ Add a 'Home' link to the page list

Enter a comma-separated list of page IDs to exclude, e.g. 5,9,27

Categories: ☑ Include categories

☐ Display post count for categories

Enter a comma-separated list of category IDs to exclude, e.g. 5,9,27

Tags: ☐ Include tags

☐ Display post count for tags

☐ Display as tag cloud

Enter a comma-separated list of tag IDs to exclude, e.g. 5,9,27

Archives: ☑ Include archives

☐ Display post count for archives

Posts: ☑ Include posts

☐ Display comment count for posts

Enter a comma-separated list of post IDs to exclude, e.g. 5,9,27

Enter a comma-separated list of category IDs to exclude, e.g. 5,9,27

Like This Plugin?

If you like this plugin, why not do any or all of the following:

- Link to it so other people can find it.
- Give it a 5-star rating on WordPress.
- Show your appreciation by making a donation.

Make A Donation!

Building and maintaining this plugin has cost me countless hours of work. If you like it, why not donate a token of your appreciation!

Need Support?

If you have any problems with this plugin or ideas for improvements or new features, please post about them on the WordPress support forums or the plugin's homepage.

Styling Issues?

The sitemap is displayed using your theme's own default styles, so it should blend seamlessly into the rest of your site.

If you're unhappy with the way your theme is displaying the sitemap, however, you can find tips on tweaking its formatting on the plugin's homepage.

How it works...

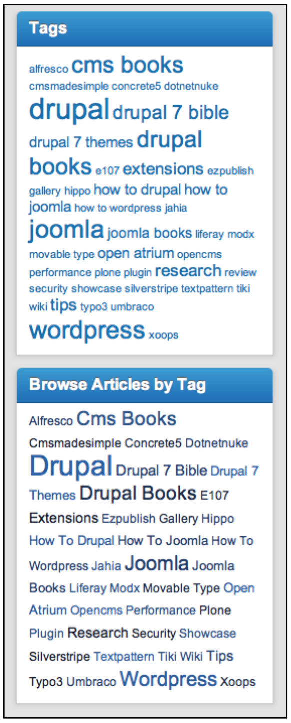
This plugin works by tapping into the power of WordPress shortcodes. When this plugin is installed and activated, it allows you to use the shortcode wherever you see fit. In this example, we placed the shortcode in a dedicated page, but you could put it inside any content area, or even into a dedicated template.

Unlike some other plugins that rely on shortcodes, there are no parameters for you to configure manually. Instead, all configuration is done through the HTML Sitemap configuration settings, discussed above.

Creating a better tag cloud

The default installation of WordPress comes with a simple tag cloud widget. The standard widget has virtually no configuration options. If you wish to have a tag cloud with more functionality, or even simply more options for styling and presentation, then you will need to explore other options. Fortunately there are a couple of tag cloud plugins. Our favorite is Ultimate Tag Cloud, which is the subject of this recipe.

The following screenshots show you the default tag cloud (at top) contrasted with the output of the Ultimate Tag Cloud plugin (at bottom):



Tags

alfresco cms books
cmsmadesimple concrete5 dotnetnuke
drupal drupal 7 bible
drupal 7 themes drupal
books e107 extensions ezpublish
gallery hippo how to drupal how to
joomla how to wordpress jahia
joomla joomla books liferay modx
movable type open atrium opencms
performance plone plugin research review
security showcase silverstripe textpattern tiki
wiki tips typo3 umbraco
wordpress xoops

Browse Articles by Tag

Alfresco Cms Books
Cmsmadesimple Concrete5 Dotnetnuke
Drupal Drupal 7 Bible Drupal 7
Themes Drupal Books E107
Extensions Ezpublish Gallery Hippo
How To Drupal How To Joomla How To
Wordpress Jahia Joomla Joomla
Books Liferay Modx Movable Type Open
Atrium Opencms Performance Plone
Plugin Research Security Showcase
Silverstripe Textpattern Tiki Wiki Tips
Typo3 Umbraco Wordpress Xoops

Getting ready

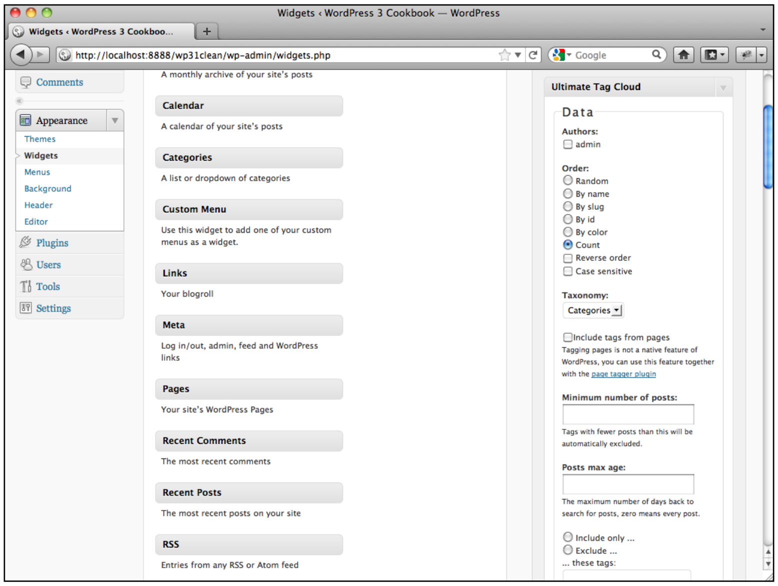
To execute this recipe, you will need to install the **Ultimate Tag Cloud** plugin. You will need to install this plugin before you can get started. Search for **Ultimate Tag Cloud** inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can learn more about the plugin by visiting the developer's website at `http://www.0x539.se/wordpress/ultimate-tag-cloud-widget/`

How to do it...

There's a bit of configuration needed for the plugin to work properly:

1. Log in to your WordPress **Dashboard**.
2. Click on the **Appearance** menu.
3. Click on the **Widgets** option.
4. Find the **Ultimate Tag Cloud** widget and drag it to the widget area where you want it to appear.
5. Click to open the options for the widget.
6. Set the Data options of the widget to determine where the tag info comes from. The following screenshot shows you part of the widget settings:


Widgets ‹ WordPress 3 Cookbook — WordPress
Widgets ‹ WordPress 3 Cookboo...
http://localhost:8888/wp31clean/wp-admin/widgets.php
Google
Comments
Appearance
Themes
Widgets
Menus
Background
Header
Editor
Plugins
Users
Tools
Settings
A monthly archive of your site's posts
Calendar
A calendar of your site's posts
Categories
A list or dropdown of categories
Custom Menu
Use this widget to add one of your custom menus as a widget.
Links
Your blogroll
Meta
Log in/out, admin, feed and WordPress links
Pages
Your site's WordPress Pages
Recent Comments
The most recent comments
Recent Posts
The most recent posts on your site
RSS
Entries from any RSS or Atom feed

Ultimate Tag Cloud

Data

Authors:
☐ admin

Order:
○ Random
○ By name
○ By slug
○ By id
○ By color
● Count
☐ Reverse order
☐ Case sensitive

Taxonomy:
Categories

☐ Include tags from pages
Tagging pages is not a native feature of WordPress, you can use this feature together with the page tagger plugin

Minimum number of posts:

Tags with fewer posts than this will be automatically excluded.

Posts max age:

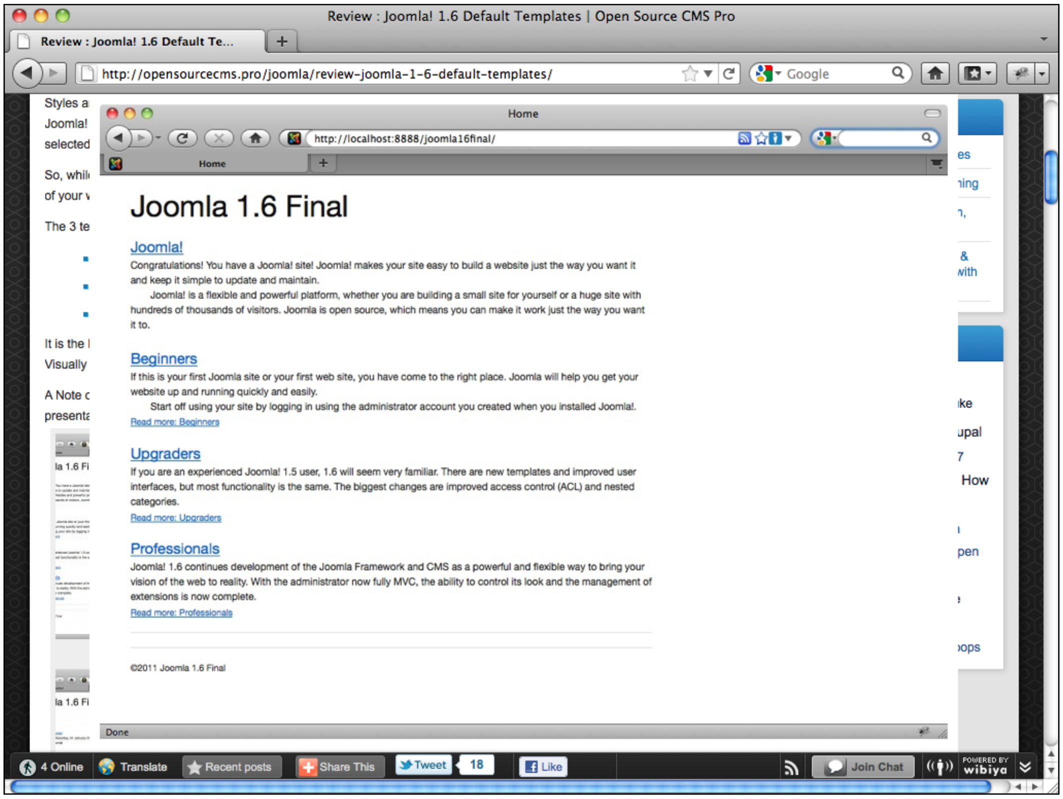
The maximum number of days back to search for posts, zero means every post.

○ Include only ...
○ Exclude ...
... these tags:

7. Set the **Appearance** options of the widget to suit your site.
8. Click on the **Save** button.

Adding lightboxes for your photos

Lightboxes are a common feature of many sites you visit these days. A lightbox works like a pop-up. It typically opens on top of the underlying page and occupies all or almost all of the screen space. When you close the pop-up, the original page is still available in the background. You can see a lightbox in action in the following screenshot:



Review : Joomla! 1.6 Default Templates | Open Source CMS Pro
Review : Joomla! 1.6 Default Te...
http://opensourcecms.pro/joomla/review-joomla-1-6-default-templates/
Google
Home
http://localhost:8888/joomla16final/
Home
Done
4 Online
Translate
Recent posts
Share This
Tweet
18
Like
Join Chat
POWERED BY wibiya

Joomla 1.6 Final

Joomla!

Congratulations! You have a Joomla! site! Joomla! makes your site easy to build a website just the way you want it and keep it simple to update and maintain.

Joomla! is a flexible and powerful platform, whether you are building a small site for yourself or a huge site with hundreds of thousands of visitors. Joomla is open source, which means you can make it work just the way you want it to.

Beginners

If this is your first Joomla site or your first web site, you have come to the right place. Joomla will help you get your website up and running quickly and easily.

Start off using your site by logging in using the administrator account you created when you installed Joomla!.

Read more: Beginners

Upgraders

If you are an experienced Joomla! 1.5 user, 1.6 will seem very familiar. There are new templates and improved user interfaces, but most functionality is the same. The biggest changes are improved access control (ACL) and nested categories.

Read more: Upgraders

Professionals

Joomla! 1.6 continues development of the Joomla Framework and CMS as a powerful and flexible way to bring your vision of the web to reality. With the administrator now fully MVC, the ability to control its look and the management of extensions is now complete.

Read more: Professionals

©2011 Joomla 1.6 Final

There are several good reasons for the popularity of lightboxes. First, for site visitors, lightboxes tend to enhance visibility by decreasing page reloads and by providing a positive aesthetic experience. For site owners, lightboxes are an easy way to implement slideshows and even forms.

In this recipe we look at how you can add lightbox functionality to your WordPress site.

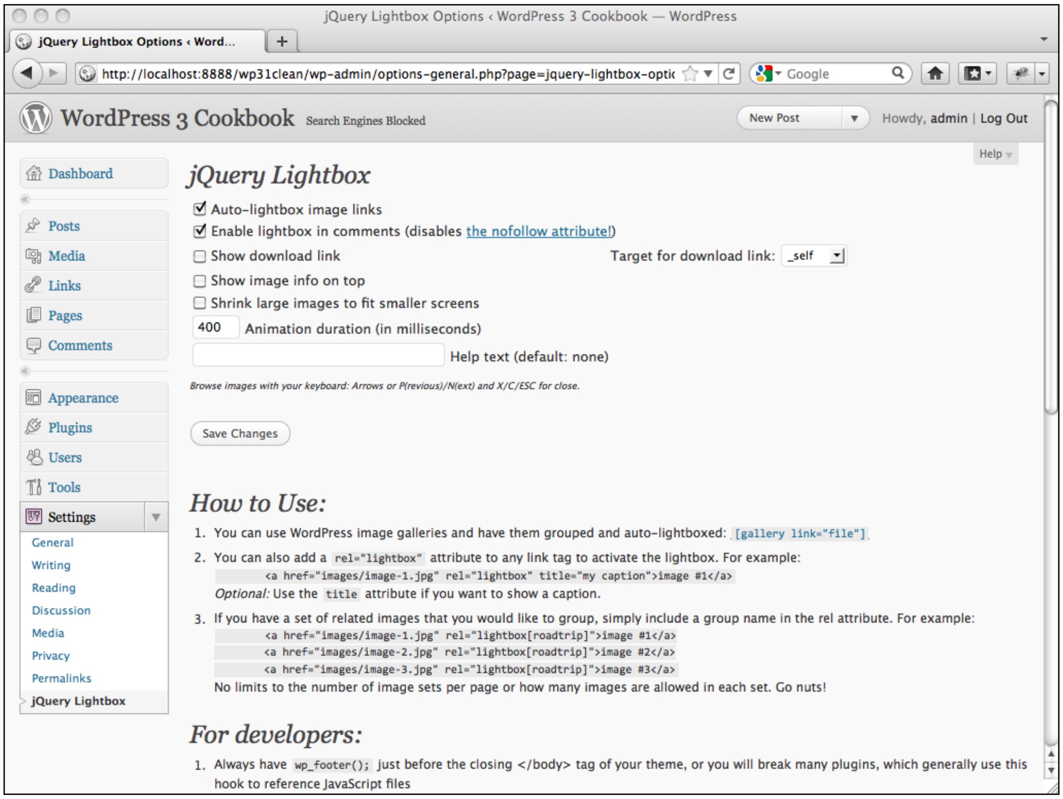
Getting ready

To execute this recipe, you will need to install the WP jQuery Lightbox plugin. You will need to install this plugin before you can get started. Search for "wp jquery lightbox" inside the **Add New Plugins** screen of your WordPress site. After you find it, click to install, and then activate it.

How to do it...

First you need to configure your lightboxes:

1. Log in to your WordPress dashboard.
2. Click on the **Settings** menu.
3. Click on the option **jQuery Lightbox**.
4. Select the configuration options shown in the following screenshot:


jQuery Lightbox Options ‹ WordPress 3 Cookbook — WordPress
jQuery Lightbox Options ‹ Word...
http://localhost:8888/wp31clean/wp-admin/options-general.php?page=jquery-lightbox-optic
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Privacy
Permalinks
jQuery Lightbox

jQuery Lightbox

☑ Auto-lightbox image links

☑ Enable lightbox in comments (disables the nofollow attribute!)

☐ Show download link — Target for download link: _self

☐ Show image info on top

☐ Shrink large images to fit smaller screens

400 Animation duration (in milliseconds)

Help text (default: none)

Browse images with your keyboard: Arrows or P(revious)/N(ext) and X/C/ESC for close.

Save Changes

How to Use:

1. You can use WordPress image galleries and have them grouped and auto-lightboxed: `[gallery link="file"]`
2. You can also add a `rel="lightbox"` attribute to any link tag to activate the lightbox. For example:

```
        <a href="images/image-1.jpg" rel="lightbox" title="my caption">image #1</a>
```

Optional: Use the `title` attribute if you want to show a caption.

3. If you have a set of related images that you would like to group, simply include a group name in the rel attribute. For example:

```
        <a href="images/image-1.jpg" rel="lightbox[roadtrip]">image #1</a>
        <a href="images/image-2.jpg" rel="lightbox[roadtrip]">image #2</a>
        <a href="images/image-3.jpg" rel="lightbox[roadtrip]">image #3</a>
```

No limits to the number of image sets per page or how many images are allowed in each set. Go nuts!

For developers:

1. Always have `wp_footer();` just before the closing </body> tag of your theme, or you will break many plugins, which generally use this hook to reference JavaScript files

5. Click the **Save Changes** button.

Next, we need to enable the lightbox functionality for our images. To do that, follow these steps:

1. Click the **Posts** menu.
2. Find a post that contains images; if you don't have one, create one. Then click to edit the post.
3. To add the lightbox functionality to an image, you need to edit the image settings.
4. Switch to HTML view.
5. Add to the image link the attribute `rel="lightbox"`
6. Update the post.

Now, if you visit the post on the front end of the website and click on the image you modified, it should pop up in a lightbox.

There's more

The jQuery Lightbox plugin also supports several other useful features; one of the most useful is the creation of slideshows.

If you wish, you can group images together to create a slideshow that can be clicked through from within the lightbox. To group images for a slideshow, simply add to the link the `rel` attribute a common group name. For example, `rel="lightbox[groupname]"`.

> It does not matter what group name you use, so long as it is used consistently on all the images you want to include in the show.

9

Managing Maintenance and Improving Security

In this chapter, we cover:

- Creating a manual backup of your database
- Creating an automatic backup with WP DB backup
- Restoring a MySQL backup
- Creating backups of your WordPress files
- Removing the WordPress version information from your theme files
- Getting rid of the administrator account
- Protecting against brute force login attempts
- Denying access to unneeded hints
- Adding another layer of protection with HTTP authentication
- Restricting access to the wp-admin directory by IP address
- Testing your site security
- Reducing SPAM by selectively blocking comment posting

Introduction

Creating and maintaining a secure website is an ongoing process. There is no one single thing you need to do, or even a definitive list of items, that will result in your site remaining secure from attack. Rather, site security is about being aware of your risk profile and taking steps to bring the risk down to acceptable levels. If your site contains user data or other sensitive, or valuable data, then you must do more than if your site is merely a single user blog.

In this chapter we go through a number of basic techniques that will harden your installation against the most common attacks and provide you with a fair degree of peace of mind. Some of the recipes are intended to block aggressors, others to simply deter them; still other recipes are aimed at decreasing SPAM on your site, as it is often used as a vector for attacks.

If you are serious about security, then the recipes in this chapter provide just a starting point. Making your WordPress installation secure is only part of the battle. If you want to increase your security significantly, then you must work with your server settings to make sure the server itself is less vulnerable. Server hardening is not covered in this chapter, as there are too many variables for us to anticipate.

Also not covered in this chapter, but worth mentioning, is the need to maintain a secure development environment and development process when you are building your site. Poor development practices often leave open avenues for later attacks. If you have inherited a site from another developer, you would do well to investigate whether you have inherited any potential problems as well.

Creating a manual backup of your database

If you remember only one recipe from this entire book, then this is the one! With WordPress, your site contents, comments, and parameters are all stored in a MySQL database. The database is the heart of your site and you don't want to run the risk that something might happen to it. Having backups of your database is not just important, it's essential.

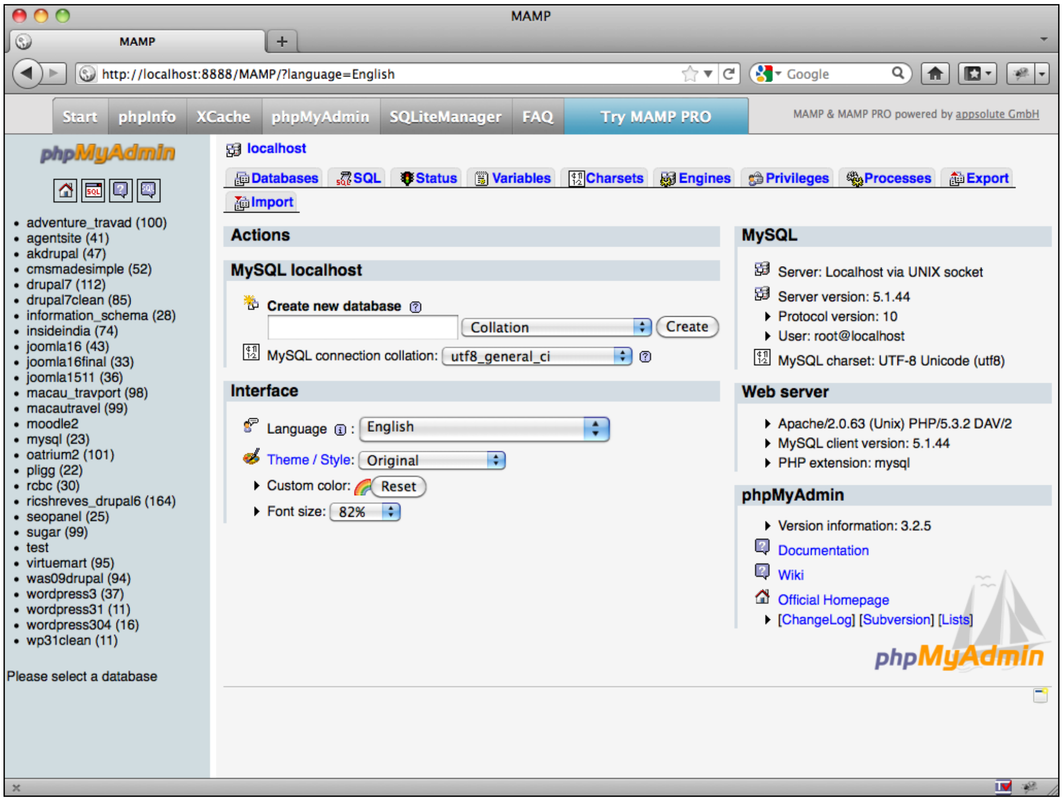
While all your site files are kept on the server, all of your WordPress data (posts, pages, categories, comments, and so on) are saved in your site's MySQL database. Without the database, the only thing you'll get is an empty theme. In this recipe, we're going to manually create a backup of your MySQL database using phpMyAdmin.

Getting ready

To complete this recipe, you must have phpMyAdmin installed on your web server. Typically this is not a problem; most web hosts have phpMyAdmin installed by default. Nonetheless, don't simply assume it is installed. Consult your hosting provider's documentation to see whether it is installed (and enabled) on your server.

How to do it...

1. Log in to your **phpMyAdmin** installation (the URL depends on how things are configured on your server. For most people you will access this through your web hosting control panel. If you don't know how to find it, then contact your web host's support team). The following screenshot shows the entry screen in a typical **phpMyAdmin** installation:


MAMP
MAMP
http://localhost:8888/MAMP/?language=English
Google
Start
phpInfo
XCache
phpMyAdmin
SQLiteManager
FAQ
Try MAMP PRO
MAMP & MAMP PRO powered by appsolute GmbH
phpMyAdmin
adventure_travad (100)
agentsite (41)
akdrupal (47)
cmsmadesimple (52)
drupal7 (112)
drupal7clean (85)
information_schema (28)
insideindia (74)
joomla16 (43)
joomla16final (33)
joomla1511 (36)
macau_travport (98)
macautravel (99)
moodle2
mysql (23)
oatrium2 (101)
pligg (22)
rcbc (30)
ricshreves_drupal6 (164)
seopanel (25)
sugar (99)
test
virtuemart (95)
was09drupal (94)
wordpress3 (37)
wordpress31 (11)
wordpress304 (16)
wp31clean (11)
Please select a database
localhost
Databases
SQL
Status
Variables
Charsets
Engines
Privileges
Processes
Export
Import
Actions
MySQL localhost
Create new database
Collation
Create
Interface
Custom color:
Reset
MySQL
Web server
phpMyAdmin
Documentation
Wiki
Official Homepage
[ChangeLog] [Subversion] [Lists]
phpMyAdmin

MySQL connection collation: utf8_general_ci

Language: English

Theme / Style: Original

Font size: 82%

Server: Localhost via UNIX socket
Server version: 5.1.44
Protocol version: 10
User: root@localhost
MySQL charset: UTF-8 Unicode (utf8)

Apache/2.0.63 (Unix) PHP/5.3.2 DAV/2
MySQL client version: 5.1.44
PHP extension: mysql

Version information: 3.2.5

2. Select the database that is being used by your WordPress site.
3. Click on the **Export** tab on the horizontal menu near the top of the screen.
4. Next, scroll to the bottom of the page and select the option **Save as file**. If you wish, you can choose to compress the backup. **Gzipped** is a good option and widely compatible.
5. Click on the **Go** button. Wait a few seconds and your browser will ask you to download the backup. Keep it in a secure place, like a USB drive.

> The export option is one of the numerous things that phpMyAdmin can do for you. Creating manual backups using the command line is easy as well, but most people will find it a bit more difficult than creating backups from within phpMyAdmin.

There's more...

If you're interested in having a plugin that can perform backups for you, and can even schedule automatic backups, then you should refer to the next recipe.

See also

- *Creating an automatic backup with WP DB Backup*
- *Creating backups of your WordPress files*

Creating an automatic backup with WP DB Backup

While the recipe above explains how to create a backup manually, you can set up automatic backups using a third-party plugin. The WP DB Backup plugin is one of several that offer the required functionality. This particular backup provides not only automated backup, but also several additional useful options, including the ability to schedule regular backups.

Getting ready

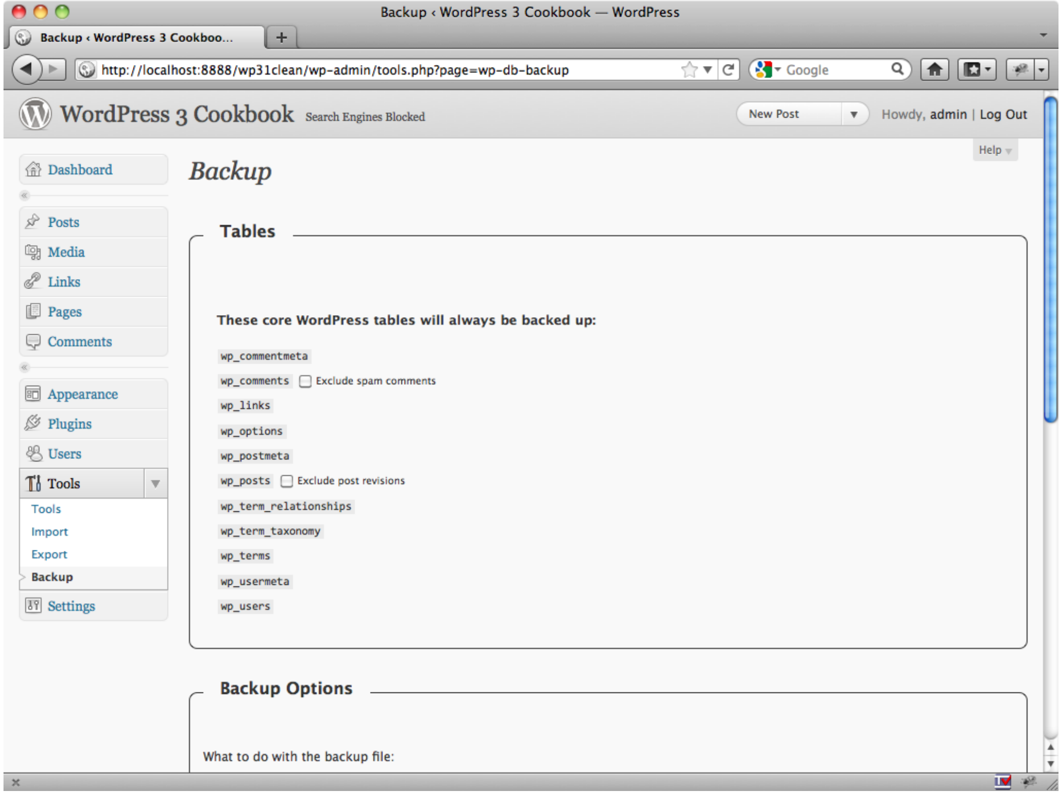
WP DB Backup is a third-party plugin. Accordingly, you must first install and enable this plugin on your WordPress site. Search for wp db backup inside the **Add New Plugins** screen of your WordPress site. After you find it, click on it to install, and then activate it.

> You can visit the plugin's home page by going to `http://austinmatzko.com/wordpress-plugins/wp-db-backup/`

How to do it...

1. Open the **Tools** menu from inside your WordPress **Dashboard**.

2. Select the option **Backup**. The WP DB **Backup** screen will load in your browser, as seen in the following screenshot:


Backup ‹ WordPress 3 Cookbook — WordPress
Backup ‹ WordPress 3 Cookboo...
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Users
Tools
Tools
Import
Export
Backup
Settings

http://localhost:8888/wp31clean/wp-admin/tools.php?page=wp-db-backup

Backup

Tables

These core WordPress tables will always be backed up:

wp_commentmeta
wp_comments ☐ Exclude spam comments
wp_links
wp_options
wp_postmeta
wp_posts ☐ Exclude post revisions
wp_term_relationships
wp_term_taxonomy
wp_terms
wp_usermeta
wp_users

Backup Options

What to do with the backup file:

3. Under the section **Backup Options**, select the option **Download to your computer**.
4. Click on the **Backup now!** button. You'll see a progress bar that indicates how the backup is proceeding. Don't refresh the page or click the **Back** button, as this will cause your backup to fail.
5. Your browser will ask you if you want to download the backup. Click on the **Yes** button and save the backup to your computer.

How it works...

While the popular phpMyAdmin web application allows you to create manual backups of any website, the WP DB Backup plugin goes further, letting you create a database backup without leaving your WordPress dashboard.

Using the cron functions of WordPress, WP DB Backup is also able to schedule automatic backups for you, which isn't possible with phpMyAdmin.

There's more...

While creating backups on demand is a cool option, it is even more secure to schedule automatic backups. WP DB Backup can easily do this. Use the **Scheduled Backup** section to create periodic updates – you can even have them automatically e-mailed to the address of your choice.

See also

- *Creating manual backups of your database*
- *Creating backups of your WordPress files*

Restoring a MySQL backup

Now that you know how to create a backup of your site database, you should also know how to restore it when needed. A restore can be done in whole or in part, but for this discussion we will assume that you are restoring the whole database.

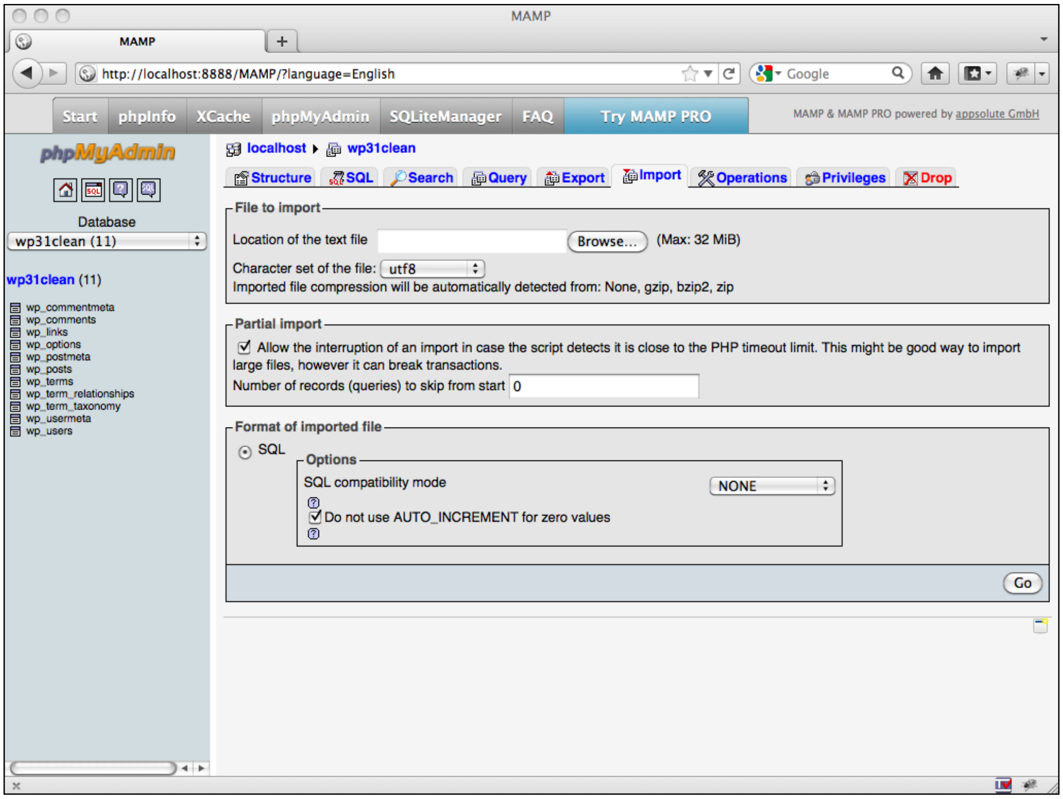
Getting ready

For this recipe, you need a backup of your site's database and access to the **phpMyAdmin** installation on your web server.

> This is for emergencies only. If you want to test this process, then you should install WordPress locally and perform your tests on the local installation; don't test this out on a live production site!

How to do it...

1. Log in to **phpMyAdmin** and select your WordPress database.
2. Click on the **Import** tab from the horizontal menu near the top of the page. The following screenshot shows the **Import** screen:


MAMP
MAMP
http://localhost:8888/MAMP/?language=English
Google
Start
phpInfo
XCache
phpMyAdmin
SQLiteManager
FAQ
Try MAMP PRO
MAMP & MAMP PRO powered by appsolute GmbH
phpMyAdmin
Database
wp31clean (11)
wp31clean (11)
wp_commentmeta
wp_comments
wp_links
wp_options
wp_postmeta
wp_posts
wp_terms
wp_term_relationships
wp_term_taxonomy
wp_usermeta
wp_users
localhost
wp31clean
Structure
SQL
Search
Query
Export
Import
Operations
Privileges
Drop
Browse...
Go

File to import

Location of the text file (Max: 32 MiB)

Character set of the file: utf8

Imported file compression will be automatically detected from: None, gzip, bzip2, zip

Partial import

☑ Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. This might be good way to import large files, however it can break transactions.

Number of records (queries) to skip from start 0

Format of imported file

● SQL

Options

SQL compatibility mode NONE

☑ Do not use AUTO_INCREMENT for zero values

3. Click on the **Browse** button and select the backup file on your hard drive.
4. Once done, scroll down the page and click on the **Execute** button.

The database backup is imported into your database and the existing data in the database has been replaced.

How it works...

When you import a MySQL backup with phpMyAdmin, your database is truncated (which means that the database becomes empty), and the previous data is replaced by the data contained in the backup. If changes have been made to the content of your site since your last backup, they will be lost. This is why you should create backups frequently. The more frequently you create backups, the less data you may lose.

Creating backups of your WordPress files

As we have seen earlier, having backups of your WordPress database is really important, but what about your files? Core files aren't really a problem because you can download them when you want from `www.wordpress.org`. However, when you're writing posts, you often upload pictures, and maybe even modify your theme; those files are not kept in the database, but rather in the directories on your server. Accordingly, we need to be able to make copies of those files as well in order to create a complete backup of your site.

Getting ready

This recipe is very easy to achieve. You will need an FTP program, access to your server, and some space on your hard drive or USB device to store the backup files.

How to do it...

1. Open your favorite FTP program, and connect to your site's web host.
2. Select all of your files, including the `.htaccess` file, and copy them into a directory on your local hard drive.

That's all there is to it! Now, if for some reason the WordPress files on your server are lost or corrupted, then you just have to open the directory, and copy anything from it to the root of your site's web host.

How it works...

There is really nothing hard here. In this recipe, you created a directory on your computer and then copied the files from your server to this directory, using an FTP program. The result is a backup of your files on your computer that can be used to replace your WordPress files, if any kind of problem occurs.

See also

- *Creating a manual backup of your database*
- *Creating an automatic backup with WP DB Backup*

Removing the WordPress version information from your theme files

By default, WordPress adds the Generator metatag to the header of your site's pages. The tag contains the WordPress version you're site is using. The problem is that, since this data is available to anyone who views the site's source, hackers who are looking for information about your site can learn what version of WordPress you're running. While that information alone is unlikely to present an opportunity for an attack on your site, it can make their job easier, and we want to block that.

Getting ready

To complete the recipe, you must access your WordPress installation, edit a file from your site, and then get it back on the server. To grab the file and replace it, you will need an FTP program or the file manager in your web hosting control panel. You can edit the file with either a web editing program or with a simple text editor.

How to do it...

1. Access your server and go to the directory that holds your theme files.
2. Open the `functions.php` file from inside your theme. If that file doesn't exist, then create it and save it inside your theme's directory.
3. Add the following code to the bottom of the `functions.php` file:

   ```
   add_filter( 'the_generator', create_function('$a',
   "return null;"));
   ```

4. Save the file.

If you reload your site's home page and view the page source, you will find that the Generator metatag is no longer displayed.

How it works...

This meta tag is added by WordPress to help them gather data about the installations on the web. While the information may be useful to the WordPress team, it is absolutely unnecessary for you or your site. We can get rid of it without losing any functionality.

The reason you need to add a filter on the`_generator()` function is to avoid displaying your WordPress version. This is done by creating a basic function that returns null instead of the WordPress version.

Getting rid of the Administrator account

By default, all WordPress sites have an Administrator account. This account is automatically created when you install WordPress. If you have not changed the default value, the username is admin. The account has administrator rights, which means that someone logged in with the Administrator account can create other user accounts, change the WordPress password, and much more.

Hackers know that every WordPress site automatically creates an Administrator account, and if they want to try to break into a WordPress site, they will typically try to target the admin user account in an attempt to crack the password. A simple, but effective way to avoid being vulnerable to this sort of attack is to create a new administrator account to replace the default admin account and then delete the original account.

Getting ready

All of the steps necessary to complete this recipe can be completed from within the WordPress administration system.

How to do it...

1. Log in to your WordPress **Dashboard**.
2. Go to the **Users** page.
3. Click on the **Add new** button.
4. Enter the details for your new account. Select the role to be **Administrator**.
5. After you have created your new account, log out.
6. Log in again, this time using the new account you just created.
7. Go to the **Users** page.
8. Find the original **Admin** account, and delete it.

How it works...

This process is necessitated by the fact that usernames in WordPress cannot be changed. Therefore, we must use our existing **Admin** account to create another account, give it Administrator rights, and then delete the original account.

See also

- *Testing your site security*

Protecting against brute force log in attempts

In the previous recipe, we covered deleting the default admin account. By deleting the admin account, you have taken one more potential avenue of attack away from hackers. However, hackers are typically smart (and persistent!) and are likely to also try other logins.

The method of attack generally involved isolating a potential username, then trying one password after another until they are successful. This type of attack is called a **brute force attack**. This recipe shows you how to install and configure a third-party plugin to help protect against this common attack vector.

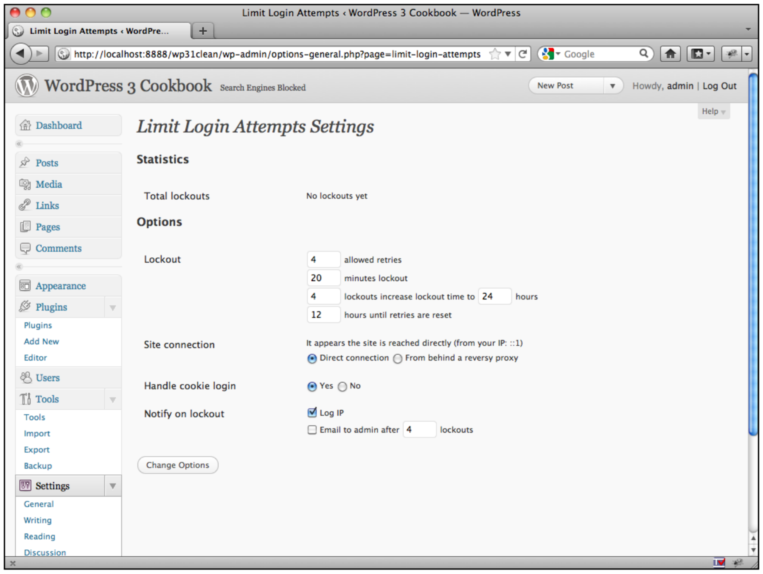
Getting ready

The recipe relies on the Limit Login Attempts plugin. You will need to install and enable the plugin, as described in *Chapter 3*, *Working with Plugins and Widgets*.

How to do it...

To configure Limit Login Attempts to protect your site, follow these steps:

1. Log in to the your WordPress **Dashboard**.
2. Under the **Settings** menu, click on the option **Limit Login Attempts**. The **Limit Login Attempts** configuration page is shown in the following screenshot:


Limit Login Attempts ‹ WordPress 3 Cookbook — WordPress
Limit Login Attempts ‹ WordPre...
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Tools
Import
Export
Backup
Settings
General
Writing
Reading
Discussion
Change Options

http://localhost:8888/wp31clean/wp-admin/options-general.php?page=limit-login-attempts

Limit Login Attempts Settings

Statistics

Total lockouts: No lockouts yet

Options

Lockout:
4 allowed retries
20 minutes lockout
4 lockouts increase lockout time to 24 hours
12 hours until retries are reset

Site connection: It appears the site is reached directly (from your IP: ::1)
● Direct connection ○ From behind a reversy proxy

Handle cookie login: ● Yes ○ No

Notify on lockout:
☑ Log IP
☐ Email to admin after 4 lockouts

3. Set the options you wish. Fewer **allowed retries** and longer values for the **minutes lockout** field are more secure.
4. Click on **Change Options.**

How it works...

A brute force attack is a method for attempting to defeat a cryptographic scheme by systematically trying a large number of possibilities. When someone attempts this on your site, the plugin in this recipe will record the IP address and timestamp of every failed login attempt. If more than a certain number of attempts are detected within a short period of time from the same IP range, then the login function is disabled for all of the requests from that range. This helps in preventing brute force password discovery.

See also

- *Adding another layer of protection with HTTP authentication*
- *Restricting access to the wp-admin directory by IP address*

Denying access to unneeded hints

Sometimes WordPress can be a bit more helpful than you might want. One example is the system's error messages on failed logins. By default, if a login fails, the system will tell the user whether the problem lies with the username they entered or if it was the password. While this is helpful to your users, it is also very helpful to anyone trying to hack into your account by compromising a user account. With the default error messages in place, the hacker will learn whether they need to concentrate their efforts on cracking the username of the password.

In this recipe we show you how to block this unwanted hint and replace it with a generic error message.

Getting ready

This recipe works with the `user.php` file, so you will need access to your WordPress installation on your server and the ability to edit this file type.

How to do it...

1. Access your WordPress installation on the server.
2. Find the `user.php` file.

3. First, find the following line of code:

```
if ( !$userdata )
   return new WP_Error('invalid_username',
     sprintf(__('<strong>ERROR</strong>: Invalid username.
      <a href="%s" title="Password Lost and Found">Lost
        your password</a>?'),
          site_url('wp-login.php?action=lostpassword',
           'login')));
```

4. Edit the line as follows:

```
if ( !$userdata )
   return new WP_Error('invalid_username',
      sprintf(__('<strong>ERROR</strong>: Please try again.
         <a href="%s" title="Password Lost and Found">Lost your
           password</a>?'), site_url
            ('wp-login.php?action=lostpassword', 'login')));
```

5. Next, find the following line of code:

```
if ( !wp_check_password($password, $userdata->user_pass,
$userdata->ID) )
    return new WP_Error( 'incorrect_password', sprintf
      ( __( '<strong>ERROR</strong>: The password you entered for
        the username <strong>%1$s</strong> is incorrect.
          <a href="%2$s" title="Password Lost and Found">Lost your
             password</a>?' ),
       $username, site_url( 'wp-login.php?action=lostpassword',
        'login' ) ) );
```

6. Edit the line as follows:

```
if ( !wp_check_password($password, $userdata->user_pass,
$userdata->ID) )
   return new WP_Error( 'incorrect_password', sprintf( __(
     '<strong>ERROR</strong>: Please try again. <a href="%2$s"
         title="Password Lost and Found">Lost your password</a>?'
),
   $username, site_url( 'wp-login.php?action=lostpassword',
     'login' ) ) );
```

7. Save the file.

How it works...

This technique simply changes the warning text of the two error messages to strip out the specific instructions that can aid a hacker. We've removed the hints and replaced them with a generic message (`Please try again.`).

Adding another layer of protection with HTTP authentication

When it comes to blocking brute force attacks, two layers of protection are better than one. In this recipe, we look at how to add another barrier to brute force attacks by using a third-party plugin named AskApache Password Protection.

The plugin makes it easy for you to set up HTTP authentication on your WordPress site. HTTP authentication is one of the most effective methods for protecting your site against brute force attacks, though it does add a bit of inconvenience for you as well; with HTTP authentication in place, you have to log in twice before you can access the WordPress Dashboard.

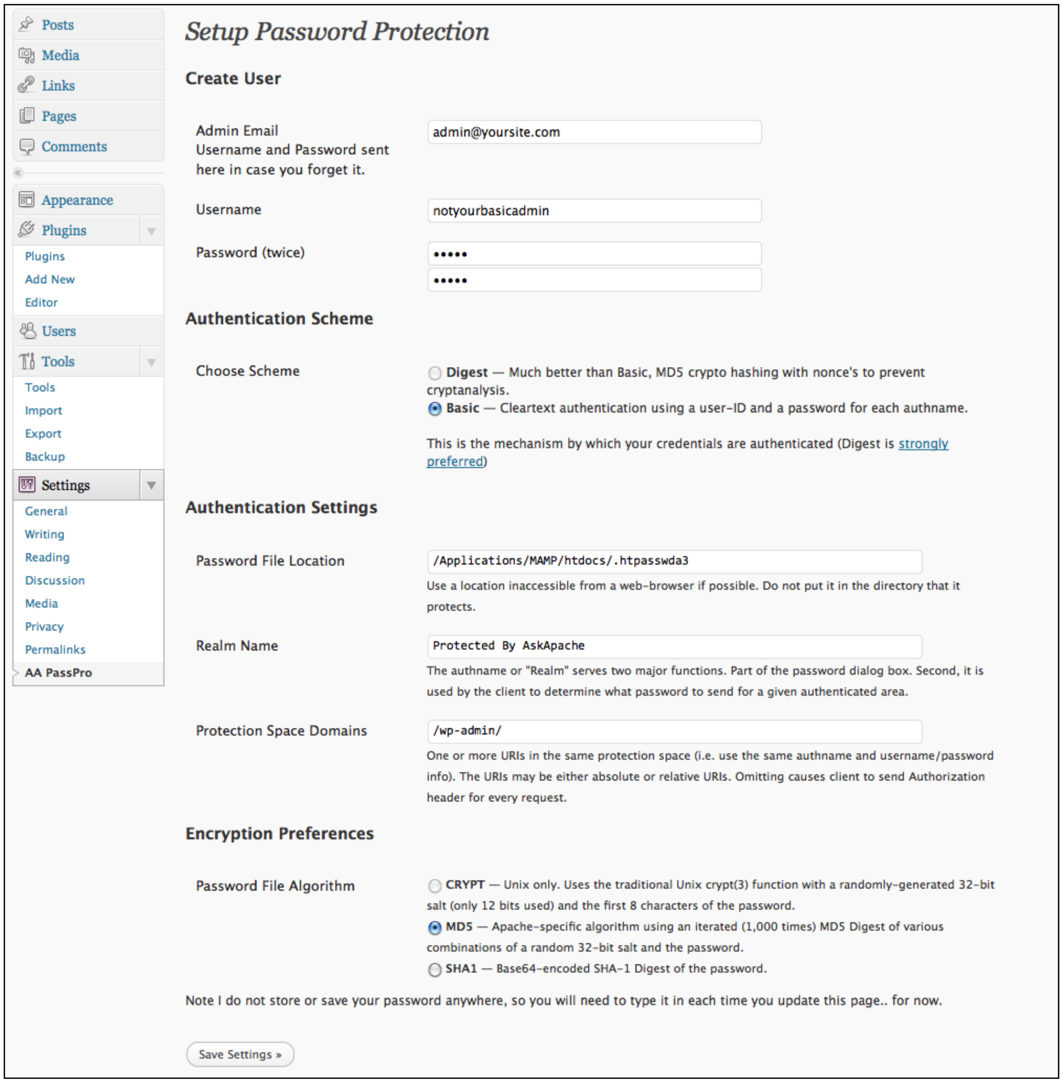
Getting ready

The recipe relies on the AskApache Password Protect plugin. You will need to install and enable the plugin, as described in *Chapter 3*, *Working with Plugins and Widgets*.

> You can learn more by visiting the plugins homepage at `http://www.askapache.com/wordpress/htaccess-password-protect.html`

How to do it...

1. Once the plugin is installed and enabled, log in to your WordPress **Dashboard**.
2. Go to **Settings | AA PassPro**.
3. If this is your first visit to this page, you will need to go through verification to make sure that the plugin will work with your server. The verification process is handled automatically by the plugin; you simply need to click on the button at the bottom of the page. Once you have completed verification, then you can move on to set up the plugin. The setup page for the plugin is shown in the following screenshot:



Setup Password Protection

Create User

Admin Email Username and Password sent here in case you forget it.	admin@yoursite.com
Username	notyourbasicadmin
Password (twice)	••••• •••••

Authentication Scheme

Choose Scheme

○ **Digest** — Much better than Basic, MD5 crypto hashing with nonce's to prevent cryptanalysis.

● **Basic** — Cleartext authentication using a user-ID and a password for each authname.

This is the mechanism by which your credentials are authenticated (Digest is strongly preferred)

Authentication Settings

Password File Location: `/Applications/MAMP/htdocs/.htpasswda3`

Use a location inaccessible from a web-browser if possible. Do not put it in the directory that it protects.

Realm Name: `Protected By AskApache`

The authname or "Realm" serves two major functions. Part of the password dialog box. Second, it is used by the client to determine what password to send for a given authenticated area.

Protection Space Domains: `/wp-admin/`

One or more URIs in the same protection space (i.e. use the same authname and username/password info). The URIs may be either absolute or relative URIs. Omitting causes client to send Authorization header for every request.

Encryption Preferences

Password File Algorithm

○ **CRYPT** — Unix only. Uses the traditional Unix crypt(3) function with a randomly-generated 32-bit salt (only 12 bits used) and the first 8 characters of the password.

● **MD5** — Apache-specific algorithm using an iterated (1,000 times) MD5 Digest of various combinations of a random 32-bit salt and the password.

○ **SHA1** — Base64-encoded SHA-1 Digest of the password.

Note I do not store or save your password anywhere, so you will need to type it in each time you update this page.. for now.

4. On the setup page, define your password settings. Pick a **Username**, a **Password**, and a message to be displayed on the authentication pop-up. The **Username** and **Password** have nothing to do with WordPress so they don't need to match your WordPress login ID and password.
5. Once done, click on the **Save Settings** button.

6. On the next screen, you can manage security modules. A security module is basically a piece of code dedicated to performing a particular action inserted in your `.htaccess` file. Use of the modules is optional, but can increase the security on your site. You may want to consider enabling at least the following modules: Password Protect `wp-login.php`; Password Protect `wp-admin`; BAD Content Type; Stop Hotlinking.

How it works...

Once activated and configured, the AskApache plugin generates a server-side password verification using the WordPress `.htaccess` file. When someone tries to access the `wp-login.php` page, a pop-up window created by the server will be launched. The user must input the proper username and password to proceed. With no password, the person attempting to view the `wp-login.php` file will not be able to see it. Note that while this same protection can be implemented manually, the AskApache Password Protection plugin does simplify implementation plus give you additional options with the security modules, making it a good choice for most site owners.

See also

- *Protecting against brute force login attempts*
- *Restricting access to the wp-admin directory by IP address*

Restricting access to the wp-admin directory by using the IP address

A very radical, but effective, solution to protect your wp-admin directory from brute force attacks, as well as any kind of intrusion, is to restrict access to this directory to a single IP address—yours.

Getting ready

Before applying this recipe, you need to make sure that you're using a static IP address. To do so, ask your **Internet Service Provider** (**ISP**). This recipe can't be achieved if you're using dynamic IP addresses.

How to do it...

1. The first thing to do is to find out your IP address. There are many ways to obtain it, but the simplest is to go to `http://whatsmyip.org`. When you visit the site, your IP address will be displayed on the page; make a note of it.

2. Next, create a file named `.htaccess` on your computer and enter the following lines in it. Do not edit the `.htaccess` file located at the root of your WordPress install.

   ```
   AuthUserFile /dev/null
   AuthGroupFile /dev/null
   AuthName "Example Access Control"
   AuthType Basic
   <LIMIT GET>
   order deny,allow
   deny from all
   allow from xx.xx.xx.xx
   </LIMIT>
   ```

   On the next to last line of the preceding code, replace xx.xx.xx.xx. with the IP address that you wish to allow access to the `wp-admin` directory.

3. Save the `.htaccess` file on the `wp-admin` directory of your WordPress site.

How it works...

In this recipe, we have used the authentication functionalities of the Apache web server to restrict access to the wp-admin directory to a specific IP address. Any IP address that isn't listed in the `.htaccess` file, will only see a **403 Forbidden** error.

There's more...

In the steps above, we have shown you how to restrict the wp-admin directory to a single IP address. However, in the case of a multiauthor site, or if you need to access the admin system from multiple locations, you may need to authorize more than just one IP address.

Allowing access to more than one IP

Use the following syntax in the `.htaccess` file (one IP address per line) to authorize multiple IP addresses:

```
AuthUserFile /dev/null
AuthGroupFile /dev/null
AuthName "Example Access Control"
AuthType Basic
<LIMIT GET>
    order deny,allow
    deny from all
    allow from xx.xx.xx.xx
    allow from xx.xx.xxx.xx
    allow from xx.xx.xxx.xx
    allow from xx.xx.xx.xx
</LIMIT>
```

See also

- *Protecting against brute force login attempts*
- *Adding another layer of protection with HTTP authentication*

Testing your site security

One of the more challenging aspects of site security is identifying potential security risks. Luckily, there are tools to help you in the quest for a secure site. In this recipe, we're going to show you how to use the WP Security Scan plugin to scan your site, get a listing of security problems, and fix them.

Getting ready

The recipe relies on the WP Security Scan plugin. You will need to install and enable the plugin, as described in *Chapter 3*, *Working with Plugins and Widgets*.

Before you begin making any modifications with the plugin, create a backup of both your database and files, as described earlier in this chapter.

How to do it...

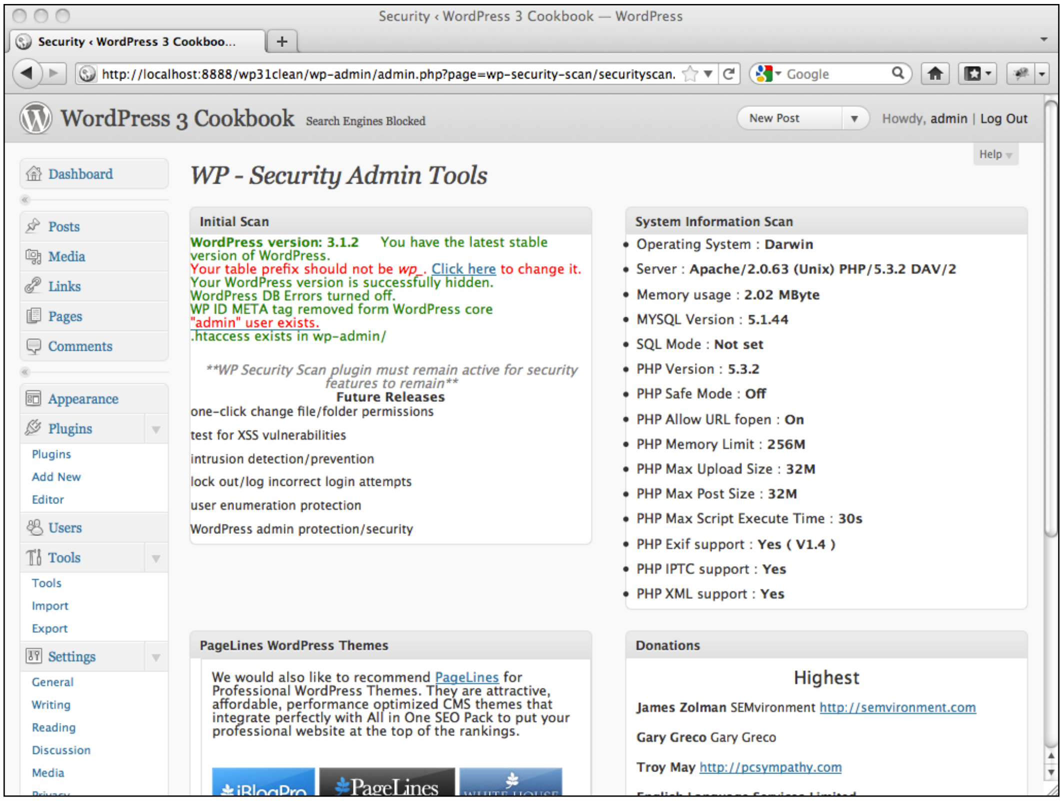
Once installed, the plugin will analyze the following:

- passwords
- file permissions
- database security
- version hiding
- WordPress admin protection and security
- WP Generator META tag

The plugin will provide recommendations for settings of each of the items above and, in some cases, allow you to implement the changes directly from within the plugin's dashboard.

After installation, the plugin adds a new option to your WordPress dashboard. To access the plugin's functionality, click on the tab labeled **Security**. This tab contains a submenu composed of the following five items:

1. **Security**: Click on this option to gain an overview of security-related information about your site and your server configuration, as you can see in the next screenshot. It also provides a particularly interesting tool, which is the database prefix switcher. If your tables' prefixes are the default `wp_`, then you'll see a prompt asking if you'd like to change your database prefix. (Having `wp_` as a prefix is a potential security risk, because many people use it. Therefore, this will be the first thing a hacker will try in order to hijack your site.) The **Security** tab will also let you know whether the WordPress version is visible and if so, will allow you to hide it. The plugin will check for an `.htaccess` file in your `wp-admin` directory and will also verify whether an **Admin** account exists.


Security ‹ WordPress 3 Cookbook — WordPress
Security ‹ WordPress 3 Cookboo...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=wp-security-scan/securityscan.
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Tools
Import
Export
Settings
General
Writing
Reading
Discussion
Media
iBlogPro
PageLines

WP - Security Admin Tools

Initial Scan

WordPress version: 3.1.2 You have the latest stable version of WordPress.
Your table prefix should not be *wp_*. Click here to change it.
Your WordPress version is successfully hidden.
WordPress DB Errors turned off.
WP ID META tag removed form WordPress core
"admin" user exists.
.htaccess exists in wp-admin/

WP Security Scan plugin must remain active for security features to remain

Future Releases

one-click change file/folder permissions

test for XSS vulnerabilities

intrusion detection/prevention

lock out/log incorrect login attempts

user enumeration protection

WordPress admin protection/security

System Information Scan

- Operating System : **Darwin**
- Server : **Apache/2.0.63 (Unix) PHP/5.3.2 DAV/2**
- Memory usage : **2.02 MByte**
- MYSQL Version : **5.1.44**
- SQL Mode : **Not set**
- PHP Version : **5.3.2**
- PHP Safe Mode : **Off**
- PHP Allow URL fopen : **On**
- PHP Memory Limit : **256M**
- PHP Max Upload Size : **32M**
- PHP Max Post Size : **32M**
- PHP Max Script Execute Time : **30s**
- PHP Exif support : **Yes (V1.4)**
- PHP IPTC support : **Yes**
- PHP XML support : **Yes**

PageLines WordPress Themes

We would also like to recommend PageLines for Professional WordPress Themes. They are attractive, affordable, performance optimized CMS themes that integrate perfectly with All in One SEO Pack to put your professional website at the top of the rankings.

Donations

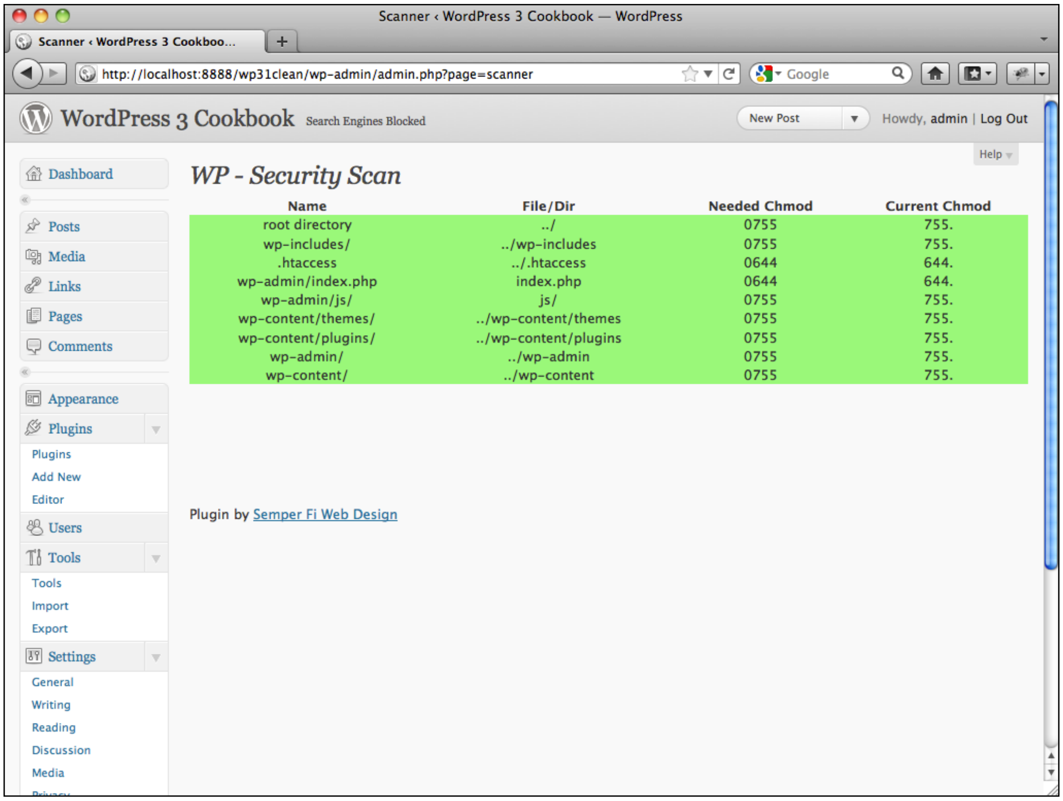
Highest

James Zolman SEMvironment http://semvironment.com

Gary Greco Gary Greco

Troy May http://pcsympathy.com

2. **Scanner**: The second option on the **Security** menu is labeled **Scanner**. The scanner feature automatically reviews your site file permissions to ensure that they are secure, as shown in the next screenshot. If one of your directories has greater access permissions than needed, it will appear in red. If any entries appear in red, you should change to the permissions indicated in the **Needed Chmod** column; this cannot be done from within the plugin, but can be done with most FTP programs.

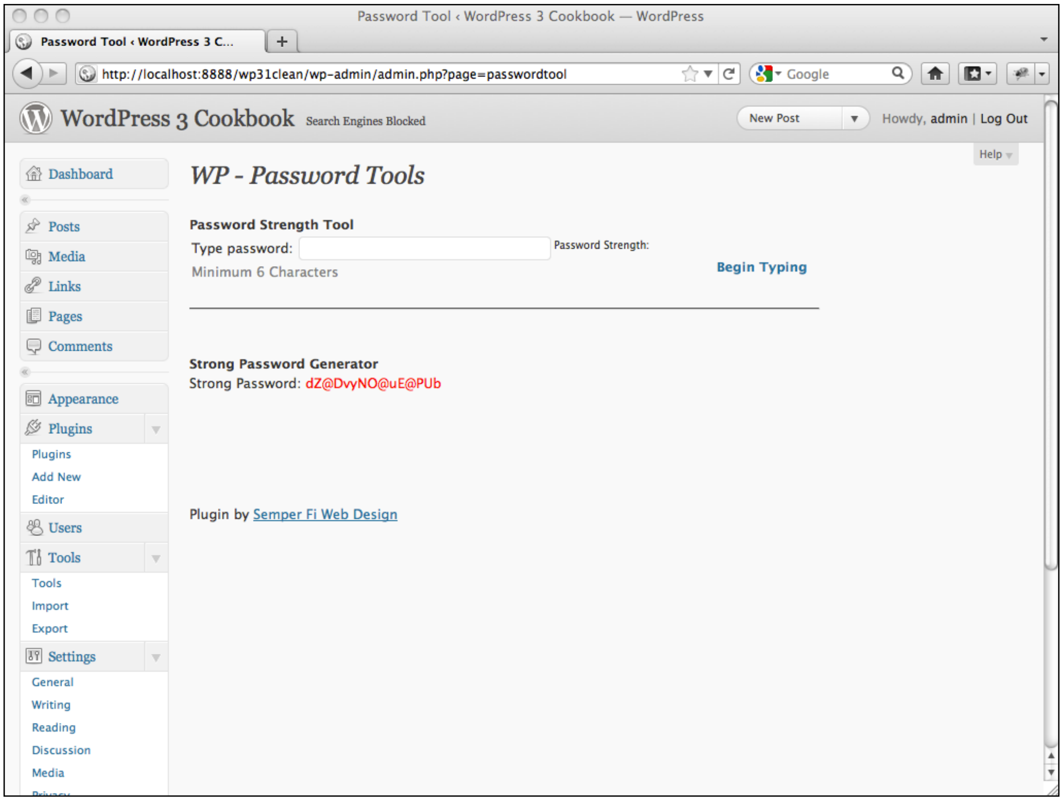

Scanner ‹ WordPress 3 Cookbook — WordPress
Scanner ‹ WordPress 3 Cookboo...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=scanner
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Tools
Import
Export
Settings
General
Writing
Reading
Discussion
Media

WP - Security Scan

Name	File/Dir	Needed Chmod	Current Chmod
root directory	../	0755	755.
wp-includes/	../wp-includes	0755	755.
.htaccess	../.htaccess	0644	644.
wp-admin/index.php	index.php	0644	644.
wp-admin/js/	js/	0755	755.
wp-content/themes/	../wp-content/themes	0755	755.
wp-content/plugins/	../wp-content/plugins	0755	755.
wp-admin/	../wp-admin	0755	755.
wp-content/	../wp-content	0755	755.

Plugin by Semper Fi Web Design

3. **Password Tool**: The **Password Tool** option provides both a utility to test the strength of your password and a password generator that can create a very strong password to help keep your user account secure. To test a password, just enter it in the **Type password** field and the utility will tell you if it's weak, average, or strong. It is best to use a strong password. To save you a bit of time, the utility also suggests a strong password for you, as you can see in the following screenshot:


Password Tool ‹ WordPress 3 Cookbook — WordPress
Password Tool ‹ WordPress 3 C...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=passwordtool
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Tools
Import
Export
Settings
General
Writing
Reading
Discussion
Media

WP - Password Tools

Password Strength Tool

Type password:

Password Strength:

Minimum 6 Characters

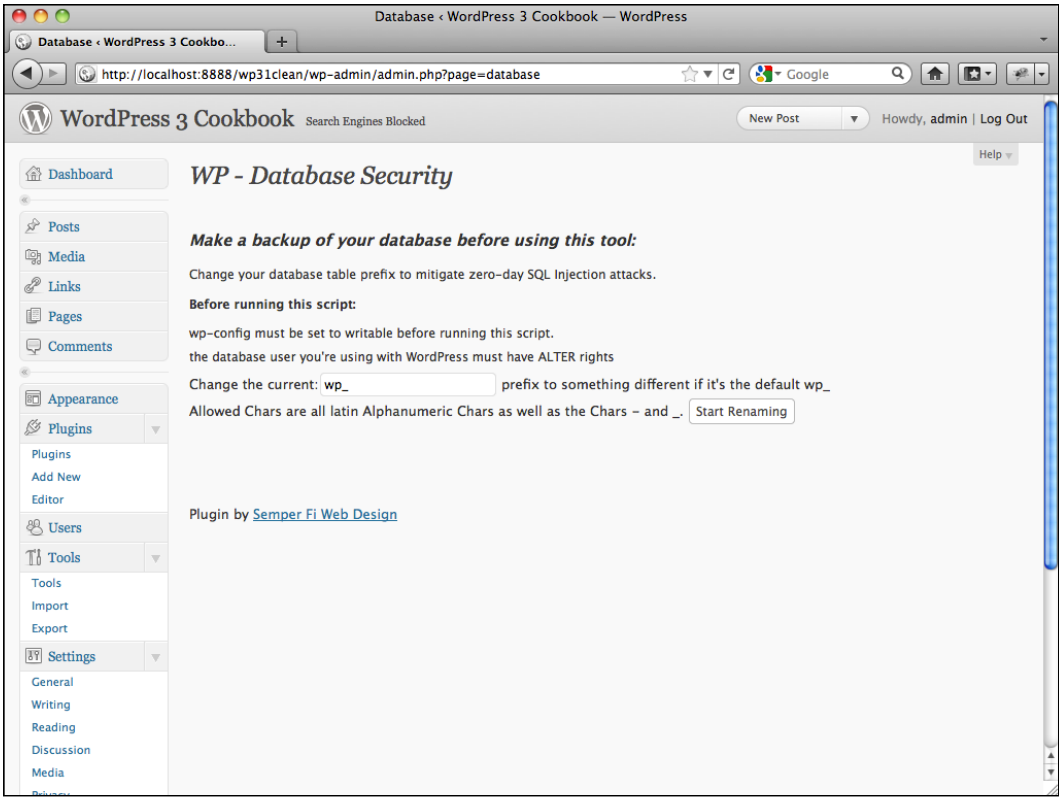
Begin Typing

Strong Password Generator

Strong Password: dZ@DvyNO@uE@PUb

Plugin by Semper Fi Web Design

4. **Database**: As noted earlier, having the default `wp_` table prefix is a potential security risk. While you can remedy this by renaming your tables using **phpMyAdmin**, WP Security Scan can do the job faster and easier. Before you begin, however, back up your database and make sure your `wp-config.php` file is writable. Once you've completed your backup and made sure the file is writable, access the **Database** option under the **Security** menu. On the image that appears, designate a new prefix and then click the **Start Renaming** button. After a short bit of time, your tables will be renamed with the selected prefix. The following screenshot shows the **Database** option dashboard:


Database ‹ WordPress 3 Cookbook — WordPress
Database ‹ WordPress 3 Cookbo...
http://localhost:8888/wp31clean/wp-admin/admin.php?page=database
Google
WordPress 3 Cookbook
Search Engines Blocked
New Post
Howdy, admin | Log Out
Help
Dashboard
Posts
Media
Links
Pages
Comments
Appearance
Plugins
Plugins
Add New
Editor
Users
Tools
Tools
Import
Export
Settings
General
Writing
Reading
Discussion
Media
Privacy

WP - Database Security

Make a backup of your database before using this tool:

Change your database table prefix to mitigate zero-day SQL Injection attacks.

Before running this script:

wp-config must be set to writable before running this script.

the database user you're using with WordPress must have ALTER rights

Change the current: wp_ prefix to something different if it's the default wp_

Allowed Chars are all latin Alphanumeric Chars as well as the Chars - and _. Start Renaming

Plugin by Semper Fi Web Design

5. **Support**: As you can expect, the **Support** tab contains links to the plugin documentation and change log.

How it works...

The WP Security Scan plugin offers an easy-to-use way to identify potential security vulnerabilities and then take steps to remedy them. While it addresses many common issues and is extremely helpful to novice users, more experienced users and those who are very security conscious will want to do more. The options given in this chapter provide further guidance to harden your WordPress installation.

See also

- *Getting rid of the administrator account*

Reducing SPAM by selectively blocking comment posting

Comment spam is not only a nuisance for site owners, but also a potential vector for malware. While the Akismet plugin included in the WordPress core is very good at helping to identity spam, it does require some management time to process the potential spam comments in the Akismet queue. Ideally, it would be better to prevent the spam submissions in the first place.

One of the most common characteristics of spam comments is the posting of links in the comment that do not point back to the domain the request came from. Most spammers don't personally go to your site to post their spam comments; instead, they use a dedicated script or software to post the comments. As a result, the comment will come from a script instead of from your site's comment form. In this recipe, we'll show you how to block comment postings that lack a valid referrer request from your site.

Getting ready

In the next section, we show you two different methods for implementing this recipe. In the first approach, we modify the `.htaccess` file. In the second approach, we use a snippet of php code placed in the `functions. Php` file.

In both techniques, you will need access to the relevant files on your server and the ability to edit those files.

How to do it...

There are two different ways to implement this recipe. The first method modifies the `.htaccess` file and rewrites rules, while the second uses php. Both methods work in a similar fashion and you can pick whichever you prefer.

To implement the `.htaccess` method, follow these steps:

1. Access your WordPress installation on the server.
2. Find the `.htaccess` file, which is located in the root of your WordPress install. Back up the file, just in case anything goes wrong.
3. Open the original file and insert the following code:

```
RewriteEngine On
RewriteCond %{REQUEST_METHOD} POST
RewriteCond %{REQUEST_URI} .wp-comments-post\.php*
RewriteCond %{HTTP_REFERER} !.*yoursite.com.* [OR]
RewriteCond %{HTTP_USER_AGENT} ^$
RewriteRule (.*) ^http://%{REMOTE_ADDR}/$ [R=301,L]
```

4. Replace `yoursite.com`, on line 4 above, with your site's URL.
5. Save the file.

Another alternative is to use php. To use this approach, follow these steps:

1. Access the server where your WordPress installation is located and find the `functions.php` file.
2. Add the following piece of code to that file:

```
   function check_referrer()
  {
       if (!isset($_SERVER['HTTP_REFERER']) ||
       $_SERVER['HTTP_REFERER'] == "")
       {
           wp_die( __('Please enable referrers in your browser,
or,
           if you\'re a spammer, bugger off!') );
       }
   }

   add_action('check_comment_flood', 'check_referrer');
```

3. Save the `functions.php` file.

How it works...

The hack consists of looking up the comment's referrer (the page from where the comment posting request comes). If the referrer doesn't exist, then it means that the request didn't come directly from your site and is likely from a script. The user is then redirected and blocked from posting.

Index

Symbols

$post() function 115
$post variable 118
$wpdb object 115
@font-face
  using 42, 43
_generator() function 287
.gif file 44
<h1> tag 48
<hgroup> tag 66
.htaccess file 286, 294, 295, 301
.htaccess method
  implementing 302
.ico file format 44
<img> tag 121
.jpg file 44
.png file 44
</urlset> tag 237

A

access
  allowing, for multiple IPs 295
  denying, to unneeded hints 290, 291
  restricting, to wp-admin directory by IP address 294
Activate link 34
Add New Plugins screen 27, 67, 92, 130, 139, 277
add_shortcode() function 126
add_shortcodes() function 126
Add to Cart buttons 214
ad manager
  new Ad Zone, creating 212
  used, for advertising space management 207-211
  working 212
Admin Bar 29
Administrator account
  removing 288
administrators
  toolbar, enabling 29, 30
admin_note() function 130
Adsense
  about 188
  integrating 188-191
  integrating, manually 192
  integrating, quicktags used 191, 192
  working 191
Advertise page
  enhancing, by adding Paypal subscriptions 206, 207
  enhancing, steps 206
advertising space
  managing, ad manager used 207-211
Akismet API key 257
Akismet plugin
  about 256, 301
  working 258
All Related Posts plugin 267
AskApache Password Protection plugin
  about 292
  URL, for homepage 292
author list
  about 166
  displaying 166, 167
  working 168
  wp_list_authors() function 168
author parameter 270
author-related information
  displaying, on posts 160, 162

author's avatar
  displaying, posts 163
  working 164
automatic backup
  creating, with WP DB Backup plugin 282, 283

B

backups
  creating, for WordPress files 286
Bing
  Webmaster Tools, using 237
Bing's webmaster tools
  using 237-239
bloginfo() function 44
breadcrumbs
  used, for navigation extension 252, 253
breadcrumbs_plus () function 254
Breadcrumbs Plus plugin
  about 253
  arguments 254, 255
  values 255
  working 254
breadcrumb trial 253
brute force attack 289
BuddyPress
  used, for creating community 169-173
  working 173
built-in Plugin Editor
  plugin files, modifying 13, 14
built-in Theme Editor
  theme files, modifying 10-12

C

cache management
  used, for performance optimization 258-261
catlink parameter 271
child theme
  about 36
  creating 36, 37
  functionality, customizing 38
  style, customizing 38
  working 37
class parameter 271
click through rates (CTRs) 194
commentators, rewarding
  Top Commentators widget used 158-160
comments
  posting, Subscribe to Comments Reloaded plugin used 155, 156
comments_open() function 55
comment spam 301
comments parameter 271
community
  about 134
  creating, BuddyPress used 169-173
conditional tags
  comments_open() function 55
  has_tag() 96
  has_tag("wordpress") function 55
  has_tag("WordPress") function 55
  in_category() 96
  is_404() function 55
  is_admin() function 55
  is_archive() function 55
  is_author() 96
  is__author() function 55
  is_category() 96
  is_category('4') function 55
  is_category() functionis_category() function, conditional tags 55
  is__date()f unction 55
  is_day() function 55
  is_front_page() 96
  is_front_page() function 55
  is_home() function 55
  is__month() function 55
  is_page() 96
  is_paged() function 55
  is_page() function 55
  is_page_template() 96
  is_page_template(about.php) 55
  is_search() function 55
  is_single() 96
  is_single() function 55
  is_sticky() 96
  is_sticky()function 55
  is_tag() 96
  is_tag() function 55
  is_time() function 55
  is__year() function 55
  pings_open() function 55
  special parameters 54
  special parameters, Array 54

  special parameters, ID 54
  special parameters, Name 54
  using, for content display control 52
  using, to control widget display 94, 95
  widget content filter, using 96, 97
  working 53, 54
configuration, Limit Login Attempts plugin 289
Configure button 28
content
  exporting 24-26
  importing 24-26
content parameter 271
core widgets
  modifying 85-89
  working 90, 91
cron functions 283
custom 404 error page
  about 63
  creating 63, 64
  working 64
custom logo
  about 45
  adding 45, 47
  starting with 45
  working 48
custom styles
  adding, to theme 66, 67

D

database
  manual backup, creating 280-282
date and time parameters, within query_posts()
  day 110
  hour 110
  minute 110
  monthnum 110
  second 110
  year 110
dateformat parameter 270
date parameter 270
Deactivate button 28
Delete Duplicate Posts plugin 230
Disable widgets on main dashboard page 204
div tag 125
Download Export File button 25
duplicate content
  avoiding, with robots.txt file 229, 230
  removing, plugin used 230, 231
dynamic_sidebar() function 81

E

editorial workflow
  about 21
  Editorial Comments field 23
  Editorial Metadata fields 23
  Notifications Subscriptions section 23
  setting up 21
  working 22, 23
Enable / Disable Dashboard Widgets 204
excerpt parameter 270
excludeposts parameter 270

F

Facebook functionality access
  enabling, to site 178-180
  working 180
favicon
  about 43
  creating 43
  integrating 43
  working 44
Featured Posts block
  about 268
  creating 268
  List Category Posts plugin, using 268
Feedburner
  integrating, into site 150, 151
  integrating, manual steps 151, 152
  working 152
form() function 85
forum
  integrating, into site 139-142
  moderators 143
  plugin, configuring 139
  skins 143
  usergroups 144
  working 143
FTP program 286

functions.php file 60, 61

G

Generator metatag 287
get_permalink() function 118
get_post() function
  using 115
get_post_meta() function 121
Google
  Webmaster Tools, using 237
Google's webmaster tools
  Diagnostics 241
  Labs 241
  site configuration 240
  using 237, 238
  working 239, 240

H

hack 302
hackers 288
has_tag(), conditional tag 96
has_tag () function 55
homepage
  static page, using 64, 65
  thumbnails, displaying 119-121
h tags structure
  using 243
HTML editor 99
HTML Sitemap Generator plugin
  about 272
  working 274
HTTP authentication
  about 292
  setting up, on WordPress site 292-294

I

id parameter 270
Import tool
  content importing, steps 24
in_category(), conditional tag 96
in_category() function 55
installing
  Jetpack 27, 28
  plugins 72
  themes 32-35
  themes manually 35
  widgets 78, 79
Install link 34
interactivity 134
Internet Service Provider (ISP) 294
is_404() function 55
is_admin() function 55
is_archive() function 55
is_author(), conditional tag 96
is_author() function 55
is_author() tag 54
is_category(), conditional tag 96
is_category() function 55
is_category tag 54
is_date() function 55
is_day() function 55
is_front_page(), conditional tag 96
is_front_page() function 55
is_home() function 55
is_month() function 55
is_page(), conditional tag 96
is_paged() function 55
is_page() function 55
is_page() tag 54
is_page_template(), conditional tag 96
is_page_template function 55
is_preview() function 55
is_search() function 55
is_single(), conditional tag 96
is_single() function 55
is_single() tag 54
is_sticky(), conditional tag 96
is_sticky() function 55
is_sticky () tag 54
is_sticky() tag 54
is_tag(), conditional tag 96
is_tag() function 55
is_time() function 55
is_year() function 55
is_year() tag 53

J

Jetpack
  about 26, 27
  installing 27, 28
  modules 26

  using 27, 28
  working 29
jQuery Lightbox plugin 278

L

lightboxes
  configuring 277, 278
  photos, adding 276
  WP jQuery Lightbox plugin, adding 277
Limit Login Attempts plugin
  configuring 289
List Category Posts plugin
  about 268
  author parameter 270
  catlink parameter 271
  class parameter 271
  comments parameter 271
  content parameter 271
  date parameter 270
  deformed parameter 270
  excerpt parameter 270
  excludepost parameter 270
  id parameter 270
  name parameter 270
  number posts parameter 270
  offset parameter 270
  orderby parameter 270
  order parameter 270
  post_parent parameter 271
  post_type parameter 271
  tags parameter 270
  template parameter 270
  thumbnails parameter 271
  working 269
login form
  about 262
  displaying 263, 264
login page
  about 48
  customizing 48-50
  functions, customizing 51, 52
  plugin's module, enabling 51
  working 51

M

manual backup
  creating, for database 280-282
media files
  managing, Media Library used 6-10
Media Library
  media files editing, steps 9
  media files, managing 6-10
meta descriptions
  about 226
  creating, for posts 227, 228
  working 227, 228
metamod directory 88
meta tag 287
moderators 143
monthnum parameter 109
multi-author WordPress blog
  author-related information, displaying on posts 161
multiple authors
  allowing, on posts 164, 165
  working 165
multiple IPs
  access, allowing for 295
multiple loops
  about 112
  using 113
  working 114
multiple page templates
  about 56
  creating 56
  functionality adding, template tags used 58
  using 56, 57
  working 57
MySQL
  backup, restoring 284, 285
MySQL backup
  restoring 284, 285
MySQL database 280

N

name parameter 270
navigation
  extending, breadcrumbs used 252, 253
  improving, paginator used 134, 136

New Ads Place screen 208
nofollow attribute
  removing, for comments posting by users 156-158
numberposts parameter 270

O

offset parameter 270
orderby parameter 270
order parameter 270
own widget
  creating 82-84
  working 85

P

pages
  tags, adding to 130
  tags, working 131
paginator
  using, for navigation improvement 134, 136
  working 137
parameters, wp_list_authors() function
  echo 168
  exclude_admin 168
  feed 168
  feed_image 168
  feed_type 168
  hide_empty 168
  html 168
  number 168
  optioncount 168
  order 168
  orderby 168
  show_fullname 168
  style 168
Password Tool option 299
Paypal subscriptions
  adding, for Advertise page enhancement 206, 207
performance
  optimizing, cache management used 258-261
Permalink Finder plugin
  working 226
permalinks
  accessing 118
  get_permalink() function, working 118
  migrating 224
  optimizing, steps 219, 220
  structure, selecting 221
  structure tags reference 222
  working 221, 224
photos
  lightboxes, adding 276
phpMyAdmin 280
pings_open() function 55
plugin files
  modifying, built-in Plugin Editor used 13, 14
plugins
  about 71
  installing 72
  installing, steps 77
plugins installation
  about 72
  automatically 74, 75
  installed plugins, deleting automatically 75, 76
  installed plugins, deleting manually 77
  starting with 72, 74
  working 75
post formats
  about 59
  aside 61
  audio 61
  chat 61
  gallery 61
  image 61
  link 61
  quote 61
  status 61
  styling, option 62
  using 60
  video 61
  working 61
post list
  background color, alternating 121, 122
post_parent parameter 271
post published today
  displaying 110, 111
  displaying, steps 111
posts
  accessing, within WordPress loop 103, 104

- accessing within WordPress loop, working 104, 105
- author-related information, displaying 160, 162
- author's avatar, displaying 163
- background color, alternating on post list 122
- by date, retrieving 108-110
- data accessing, outside WordPress loop 114, 115
- displaying, in two columns 123-125
- displaying, on posts 152, 154
- get_post_data (), using 117
- get_post(), using 115-117
- multiple authors, allowing 164, 165
- notes, adding to 128-130
- published today, retrieving 110, 111
- published year ago, retrieving 111, 112
- retrieving, from specific category 105, 107
- retrieving, within WordPress loop 102, 103
- specific number, retrieving 107, 108
- thumbnails, displaying on blog homepage 121
- thumbnails, displaying on homepage 119-121

post_type 271

print friendly pages
- about 246
- creating 246-248
- print link, styling 248
- print page, styling 248
- WP Print Friendly plugin 246
- WP Print Friendly plugin, working 248

Q

query_posts() function 107, 108

R

redirects
- adding, for changed URLs 225
- Permalink Finder plugin, using 225
- Permalink Finder plugin, working 225

register_sidebar() function 81

related posts
- about 265
- All Related Posts plugin, using 265
- displaying 265, 266

related posts plugin
- configuring 265
- feature 267
- working 266, 267

retweet button
- displaying, on posts 152, 154

robots.txt file
- used, for avoiding duplicate content 229, 230

roles
- controlling 18
- creating 19, 20
- editing 19
- working 21

RSS content
- about 146-149
- aggregating 146-149
- working 149, 150

RSS feeds
- ads, inserting 198, 199
- working 200

RSS Footer plugin 198

S

Save Changes button 99

scanner feature 298

Scheduled Backup section 284

searched text
- highlighting, in search results 137

Search engine optimization. *See* **SEO**

Search Engine plugin
- about 249
- working 251, 252

search results
- plugin, working 138
- searched text, highlightining 137

SEO
- about 217
- backlinks, getting 243
- blog, checking for XHTML validity 243
- h tags structure, using 243
- improving, SEO Ultimate plugin used 241
- keywords, using 243, 244
- permalinks optimizing, steps 220
- permalinks optimizing, working 221
- tips 242

SEO, enhancing tips
  permalinks, optimizing 223
SEO Friendly Site, creating
  permalinks, safe migration 224
SEO Ultimate plugin
  SEO, improving 242
shopping cart
  adding, to site 213-215
  working 216
Show Admin Bar 30
sidebar
  tabs, displaying 92
Sidebar Login plugin
  about 262, 263
  working 264
simple gallery
  adding, to site 174-176
  parameters 177
  working 177
site
  exposing, to search engines 218, 219
  Facebook functionality access, enabling 178-180
  Feedburner, integrating into 150, 151
  forum, integrating 139-142
  monetizing 187
  shopping cart, adding 213-215
  simple gallery, adding 174-176
  Twitter stream, integrating into 180-185
site indexing
  about 233
  enhancing, XML sitemaps used 233
site indexing enhancing, XML sitemaps used
  about 233
  sitemaps, creating 237
  standard sitemap, attributes 236
  steps 234
  working 236
  XML Sitemap Generator 234
sitemap
  data, to exclude 18
  working 18
sitemap, of site visitors
  adding 271-274
site mobile device
  working 69
  WPtouch plugin, suing 67-69
site security
  about 280
  testing 296-301
site stats
  displaying, for advertisers search 200-204
  working 204, 205
skins 143
social bookmarking buttons
  about 144
  adding, to theme 144, 145
  working 145
spam
  about 256
  Akismet plugin, using 256
  reducing, by blocking comment posting 301, 302
  stopping 256, 257
spammers 301
static page
  using, as homepage 64, 65
  working 66
Stop Hotlinking 294
structure tags, permalink
  %author% 222
  %category% 222
  %hour% 222
  hyphen (-) separator 222
  %minute% 222
  %monthnum% 222
  %post_id% 222
  %second% 222
  %year% 222
style.css file 41
Subscribe to Comments Reloaded plugin
  using, for comments 155, 156
  working 156

T

tabs, displaying on sidebar
  starting with 92
  steps 92-94
tag cloud
  about 274
  Ultimate Tag Cloud plugin, using 275, 276
tags parameter 270
template parameter 270

templates
  modifying 254
template tags
  author tags 105
  bookmark tags 105
  category tags 105
  coauthors() 165
  coauthors_firstnames() 166
  coauthors_IDs() 166
  coauthors_lastnames() 166
  coauthors_links() 166
  coauthors_nicknames() 166
  coauthors_posts_links() 165
  comment tags 105
  general tags 105
  link tags 105
  navigation menu tags 105
  post tags 105
  post thumbnail tags 105
template tags, author related
  the_author_aim 163
  the_author_description 163
  the_author_firstname 163
  the_author_lastname 163
  the_author_nickname 163
  the_author_url 163
  the_author_yim 163
the_author_aim 163
the_author_description, template tags 163
the_author_firstname, template tags 163
the_author_lastname, template tag 163
the_author_nickname, template tag 163
the_author_url, template tag 163
the_author_yim, template tag 163
theme colors
  Cascading Style Sheets, working 39
  important points 40
  modifying 38, 39
theme colors, modifying
  steps 39
Theme editor
  starting 11
theme files
  modifying, built-in Theme Editor used 10, 11, 12
  version information, removing from 287
theme fonts
  @font-face, using 42
  modifying 40, 41
  web-safe fonts 41
theme installation
  starting with 32
  steps 32-35
  Themes option 33
themes
  about 32
  colors, modifying 38
  custom styles, adding 66
  fonts, modifying 40
  installing 32-35
  installing, manually 35, 36
  single widget-ready zone, working 81
  social bookmarking buttons, adding 144, 145
  widget areas, adding 79, 80
third party services
  about 231
  pinging 232
  working 232
thumbnails parameter 271
toolbar
  enabling, for administrators 29, 30
  enabling, for users 29, 30
Top Commentators widget
  about 158, 160
  working 160
Twitter stream
  integrating, into site 180-185
  working 184, 185
Twitter Tools plugin
  twitter integrating, steps 183

U

Ultimate Tag Cloud plugin 275
Update button 100, 165
Update File button 11, 67
update() function 85
Update Media button 10
Update RSS Footer Settings button 199
Update User button 17
URL re-writing 220
usergroups
  forum 144

user permissions
  controlling 18
users
  managing 14-18
  toolbar, enabling 29, 30

V

version information
  removing, from theme files 287
visibility
  about 194, 195
  managing 195, 196
  Who Sees Ads plugin, working 197

W

web-safe fonts 41
Who Sees Ads plugin
  global options 197
  global options, click safety 197
  global options, date format 197
  global options, old post 197
  global options, regular reader 197
  working 197
widget_content filter 97
widget display
  controlling, conditional tags used 94, 95
  working 96
widget() function 85
widgets
  about 71
  core widget, modifying 85
  installing 78, 79
  own widget, creating 82-84
  placing, inside of pages 97-100
  placing, inside of posts 97-100
  shortcode use, enabling 126, 127
  shortcode use, working 127
  using 77, 78
  working 100
Widgets option 211
WordPress
  about 5
  Admin Bar 29
  administration system 5
  child theme, creating 36
  content, exporting 24
  content exporting, Export tool used 25
  content, importing 24
  content importing, Import tool used 24
  editorial workflow, setting up 21
  favicon, integrating 44
  homepage, thumbnails displaying 119-121
  Jetpack, installing 26
  Jetpack, using 26
  media files managing, Media Library used 6
  permissions, controlling 18
  plugin files modifying, built-in Plugin Editor used 13
  roles 18
  theme files modifying, built-in Theme Editor used 10
  themes 32
  toolbar for administrators, enabling 29
  toolbar for users, enabling 29
  user roles, controlling 18
  users, managing 14
  version information, removing from theme files 287
WordPress blog
  core widgets, modifying 92
  homepage, thumbnails displaying 121
WordPress Codex 36
WordPress files
  backups, creating 286
WordPress gallery shortcode 177
WordPress loop
  about 102, 103
  outsidedata posts, accessing 114, 115
  posts, accessing 103, 104
  posts accessing, working 104, 105
  posts, retrieving 102, 103
  template tags. category 105
  URL 103
Wordpress.org 67
WordPress search
  about 249
  extending 249, 251
  Search Engine plugin 249
  Search Engine plugin, working 251, 252
WordPress shortcodes
  creating 126
  starting 192
  using 125

used, for ad display 192, 193
working 194
WordPress site
access, denying to unneeded hints 290, 291
access, restricting to wp-admin directory by IP address 294
Administrator account, removing 288
automatic backup, creating with WP DB Backup plugin 282, 283
backups, creating for files 286
HTTP authentication, setting up 292-294
manual backup, creating for database 280-282
MySQL backup, restoring 284, 285
protecting, against brute force login attempts 289, 290
security, testing 296-301
spam, reducing by blocking comment posting 301, 302
version information, removing from theme files 287
WordPress theme
custom 404 error page, creating 63
favicon, integrating 44
WordPress version information
removing, from theme files 287
WP database backup
downloading 19
WP DB Backup plugin
about 282
automatic backup, creating with 282, 283
WP jQuery Lightbox plugin 277
wp_list_authors() function 168
parameters 168
WP Print Friendly plugin
about 246
working 248
WP_query() function 112, 113
WP Security Scan plugin 296, 301
WP Super Cache plugin
about 259
configuring 261
content delivery network 262
content, preloading 262
options 262
working 261
WP Super Cache Settings page 260
WPtouch plugin 67

Thank you for buying

WordPress 3 Cookbook

About Packt Publishing

Packt, pronounced 'packed', published its first book "*Mastering phpMyAdmin for Effective MySQL Management*" in April 2004 and subsequently continued to specialize in publishing highly focused books on specific technologies and solutions.

Our books and publications share the experiences of your fellow IT professionals in adapting and customizing today's systems, applications, and frameworks. Our solution based books give you the knowledge and power to customize the software and technologies you're using to get the job done. Packt books are more specific and less general than the IT books you have seen in the past. Our unique business model allows us to bring you more focused information, giving you more of what you need to know, and less of what you don't.

Packt is a modern, yet unique publishing company, which focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike. For more information, please visit our website: `www.packtpub.com`.

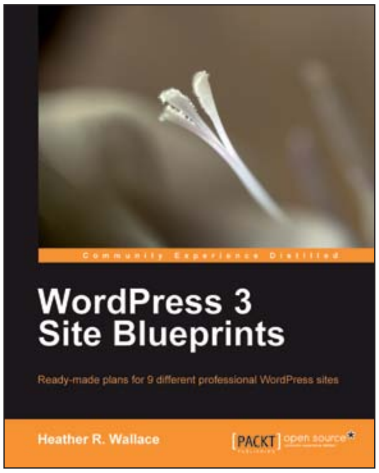
About Packt Open Source

In 2010, Packt launched two new brands, Packt Open Source and Packt Enterprise, in order to continue its focus on specialization. This book is part of the Packt Open Source brand, home to books published on software built around Open Source licences, and offering information to anybody from advanced developers to budding web designers. The Open Source brand also runs Packt's Open Source Royalty Scheme, by which Packt gives a royalty to each Open Source project about whose software a book is sold.

Writing for Packt

We welcome all inquiries from people who are interested in authoring. Book proposals should be sent to author@packtpub.com. If your book idea is still at an early stage and you would like to discuss it first before writing a formal book proposal, contact us; one of our commissioning editors will get in touch with you.

We're not just looking for published authors; if you have strong technical skills but no writing experience, our experienced editors can help you develop a writing career, or simply get some additional reward for your expertise.


Community Experience Distilled
WordPress 3 Site Blueprints
Ready-made plans for 9 different professional WordPress sites
Heather R. Wallace
PACKT open source

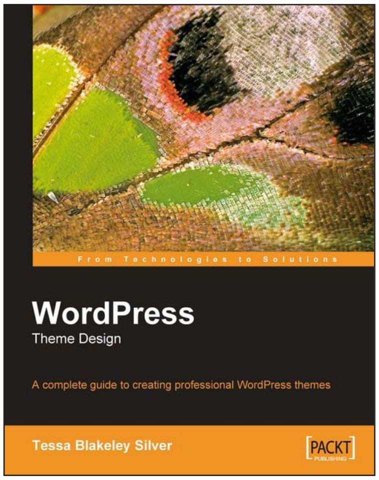
WordPress 3 Site Blueprints

ISBN: 978-1-847199-36-2 Paperback: 300 pages

Ready-made plans for 9 different professional WordPress sites

1. Everything you need to build a varied collection of feature-rich customized WordPress websites for yourself
2. Transform a static website into a dynamic WordPress blog
3. In-depth coverage of several WordPress themes and plugins
4. Packed with screenshots and step-by-step instructions to help you complete each site

---


From Technologies to Solutions
WordPress
Theme Design
A complete guide to creating professional WordPress themes
Tessa Blakeley Silver
PACKT PUBLISHING

WordPress Theme Design

ISBN: 978-1-847193-09-4 Paperback: 224 pages

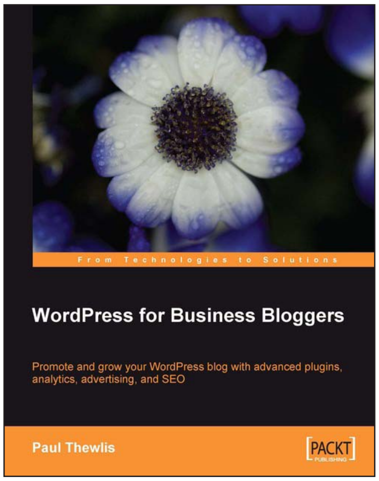
A complete guide to creating professional WordPress themes

1. Take control of the look and feel of your WordPress site
2. Simple, clear tutorial to creating Unique and Beautiful themes
3. Expert guidance with practical step-by-step instructions for theme design
4. Design tips, tricks, and troubleshooting ideas

Please check **www.PacktPub.com** for information on our titles

www.it-ebooks.info

[PACKT] open source
PUBLISHING
community experience distilled

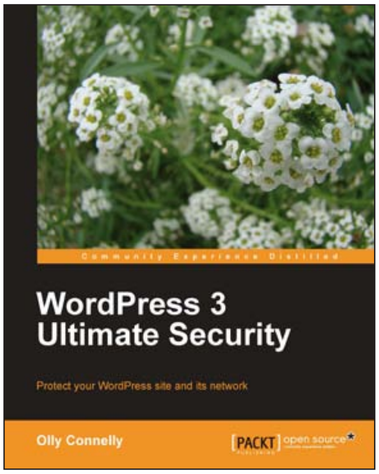

From Technologies to Solutions
WordPress for Business Bloggers
Promote and grow your WordPress blog with advanced plugins, analytics, advertising, and SEO
Paul Thewlis
PACKT PUBLISHING

WordPress for Business Bloggers

ISBN: 978-1-847195-32-6 Paperback: 356 pages

Promote and grow your WordPress blog with advanced plug-ins, analytics, advertising, and SEO

1. Gain a competitive advantage with a well polished WordPress business blog
2. Develop and transform your blog with strategic goals
3. Create your own custom design using the Sandbox theme
4. Apply SEO (search engine optimization) to your blog
5. Market and measure the success of your blog


Community Experience Distilled
WordPress 3 Ultimate Security
Protect your WordPress site and its network
Olly Connelly
PACKT PUBLISHING open source

WordPress 3 Ultimate Security

ISBN: 978-1-84951-210-7 Paperback: 408 pages

Protect your WordPress site and its network

1. Know the risks, think like a hacker, use their toolkit, find problems first – and kick attacks into touch
2. Lock down your entire network from the local PC and web connection to the server and WordPress itself
3. Find out how to back up and secure your content and, when it's scraped, know what to do to enforce your copyright

Please check **www.PacktPub.com** for information on our titles